PHD

Inspection Process Planning For Large Volume Metrology In Digital Environment

Cai, Bin

Award date:
2013

Awarding institution:
University of Bath

Inspection Process Planning For Large Volume Metrology In Digital Environment

Bin Cai

A thesis submitted for the degree of Doctor of Philosophy

University of Bath

Department of Mechanical Engineering

2012

Dedicated to Jingjin and Kellen

ABSTRACT

Nowadays, inspection process planning (IPP) for large volume metrology (LVM) attracts increasing attention in manufacturing and assembly industries such as aerospace and automotive, where large and complex assemblies and fabrications with complex surfaces are employed. Inspection is conventionally considered as a quality control manner. But there is changing shift to processes that are more related to the early design stage aiming to increase product performance and reduce costs by automation and elimination of rework. This is especially evident in the standardisation and implementation of Geometric Dimensioning and Tolerancing (GD&T) of new products and systems at the design stage.

This study proposes a GD&T based systematic framework for the IPP of LVM systems within a digital environment. Orientating to solve the “what to measure” and “how to measure” problems in IPP, the prototype system has seven functional core modules including: tolerance feature analysis, instrument selection, inspection point selection, accessibility and visibility analysis, instrument setup and configuration, clustering analysis and measurement sequencing. An optimized inspection plan is output for the designer to evaluate the product design as well as for guiding the metrologist and process planner to conduct the inspection process. Heuristic rules, evolutionary algorithms and modern computational graphic techniques have been adopted to facilitate the supported functions. Coupled with state of art metrology systems, metrology and CAD software, the framework is able to work effectively and efficiently by means of incorporating international standards and industrial best practice. It is the first attempt to successfully minimise manual activities in the planning process for LVM systems, which results in improved efficiency, enhanced decision making and a better inspection plan overall. Two case studies have been conducted to validate the functionalities of the prototype system.

ACKNOWLEDGEMENTS

Looking back, it would not have been possible to compose this doctoral thesis without the kind help and consistent support of the people around me, to only some of whom it is possible to address here.

First and foremost, I would like to express and address my deepest appreciation to my supervisor, Professor Paul G Maropoulos for his continual support and help throughout the study. You kindly provided me a belief and a freedom to explore whilst offering constant encouragement, great patience, valuable advice and guidance. It has been my greatest pleasure and honour to carry out the research under your supervision.

I would like to acknowledge the selfless guidance and help to my research provided by Dr. Yanwu Guo and Dr. Jarfar Jamshidi during their stay at University of Bath. You led me into this research area and supported me with your experience and knowledge from the beginning. It would have been much harder for me without you.

Further, I would like to thank all colleagues who shared talents, knowledge and experience with me: Per Saunders, Jody Muelaner, Zheng Wang, Olive Martin, Parag Vichare, Ben Adeline, Maria Zhang and Amir Kayani. My gratitude goes to the collaboration, contributions, discussions, comments and influences, which were provided by you throughout my research. I also wish to thank all the staff in the Department of Mechanical Engineering at the University of Bath, who have supplied me assistance and help.

Last, but by no means least, my wife Jingjin Kuai, my son Kellen Cai, my parents Yongming Cai and Xiaoli Che, my parents-in-law Ming Kuai and Lixia Mao, deserve the most special gratitude for supporting and believing in me throughout this period. Jingjin, thank you for your unceasing support during all the days we spent together.

TABLE OF CONTENTS

Abstract iii

Acknowledgements iv

Table of contents v

List of abbreviations ix

List of figures xii

List of tables xvii

Chapter 1 Introduction 1

1.1 Research background 1

1.2 Research aim, methodology and scope 3

1.2.1 Research aim 3

1.2.2 Research methodology 4

1.2.3 Research scope 5

1.3 The structure of the thesis 6

Chapter 2 Literature and state of art review 8

2.1 Introduction 8

2.2 Evolution of the IPP 9

2.2.1 Early development of IPP – feature-based approaches 9

2.2.2 Recent development of IPP- integration with CAD environment 14

2.3 Development of LVM 19

2.3.1 Laser tracker 19

2.3.2 Coherent laser radar 20

2.3.3 iGPS 22

2.3.4 Photogrammetry 22

2.4 Metrology software 23

2.5 Geometric dimensioning and tolerancing 25

2.5.1 GD&T standards 26

2.5.2 GD&T classification 26
2.6 Visibility analysis 29
2.6.1 Systems that compute approximate solution 29
2.6.2 Systems that compute exact solution 34
2.6.3 Systems utilizing graphic hardware 38
2.7 Clustering analysis 40
2.7.1 Clustering algorithm based on the discrete solution space of accessibility analysis 41
2.7.2 Clustering algorithm based on the exact solution space of accessibility analysis 45
2.8 Measurement path planning for CMMs 47
2.9 Research gaps 51
Chapter 3 Research objectives, stages, deliverables and outline of the new IPP framework for LVM systems 53
3.1 Introduction 53
3.2 Research objectives 53
3.3 Research stages and deliverables 56
3.4 Proposed framework for IPP for LVM systems 58
Chapter 4 Tolerance feature analysis 64
4.1 Introduction 64
4.2 Definition of tolerance taxonomy in the proposed system 65
4.3 GD&T extraction and inspection tasks identification 66
4.4 Summary 74
Chapter 5 Instrument selection and inspection point determination 75
5.1 Introduction 75
5.2 Proposed instrument selection process 77
5.2.1 Measurability characteristics 77
5.2.2 Proposed system 83
5.2.3 Numerical case study 98
5.2.4 Summary of instrument selection 103
5.3 Inspection point selection 103
5.3.1 Number of points 104
5.3.2 Distribution of points 104

5.4 Summary 107

Chapter 6 Visibility analysis for LVM applications 108

6.1 Introduction 108

6.2 Proposed methodology 108

6.2.1 Problem modelling 109

6.2.2 Terminology in visibility analysis 111

6.2.3 Assistant techniques for visibility analysis 121

6.2.4 Algorithm for the computation of instrument configuration space 132

6.2.5 Computation of visibility cones 134

6.3 Summary 144

Chapter 7 Clustering analysis 146

7.1 Introduction 146

7.2 Proposed clustering algorithm 147

7.2.1 Set covering problem 147

7.2.2 Clustering process modelling 149

7.2.3 Solution to SCP 152

7.2.4 Intersection of the GICSs 158

7.3 Summary 162

Chapter 8 Instrument configuration and measurement task sequencing 163

8.1 Instrument configuration 163

8.1.1 Introduction 163

8.1.2 Proposed methodology for laser tracker configuration 163

8.2 Measurement sequencing 172

8.2.1 Introduction 172

8.2.2 Proposed path planning for LVM instruments 173

8.2.3 Implementation of the genetic algorithm and experimental results 193

8.2.4 Discussion 203

8.2.5 An exploration of multiple instruments sequencing and path planning 203

8.3 Summary 206

Chapter 9 Validation of the developed methods using case studies 207

9.1 Introduction 207

9.2 Case study 1: a large-scale part with typical features and GD&T information ... 207
9.2.1 Tolerance analysis ... 208
9.2.2 Instrument selection and inspection point determination ... 211
9.2.3 Visibility analysis ... 217
9.2.4 Clustering analysis ... 222
9.2.5 Instrument setup and configuration ... 224
9.2.6 Measurement sequencing ... 225
9.3 Case study 2: an assembly wing box with given inspection points ... 229
9.3.1 Instrument selection ... 231
9.3.2 Visibility analysis ... 234
9.3.3 Clustering analysis ... 243
9.3.4 Instrument setup and configuration ... 239
9.3.5 Measurement sequencing ... 243
9.4 Summary ... 252
Chapter 10 Conclusion and future work ... 254
10.1 Conclusion and contributions ... 254
10.2 Limitations and future work ... 256
References ... 259

LIST OF ABBREVIATIONS

AABB	Axis Aligned Bounding Boxes
AC	Accessibility Cone
ADM	Absolutely Distance Measurement
API	Application Programming Interface
ASME	American Society of Mechanical Engineers
aTSP	Asymmetric Travelling Salesperson Problem
BSI	British Standard Institution
CAA	Component Application Architecture
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CCD	Charge Coupled Device
CGAL	Computational Geometry Algorithms Library
CIL	Capable Instrument List
CMM	Coordinate Measuring Machine
CSO	Configuration Space Obstacle
DM	Decision Maker
DOP	Discrete Oriented Polytope
FST	Fuzzy Set Theory
GA	Genetic Algorithm
GAC	Global Accessibility Cone
GD&T	Geometric Dimensioning and Tolerancing
GICS	Global Instrument Configuration Space

GOM	Global Occlusion Map
GPS	Geometrical Product Specifications
GPU	Graphics Processing Units
GT	Group Technology
GVC	Global Visibility Cone
GVM	Global Visibility Map
ICS	Instrument Configuration Space
IFS	Intuitionistic Fuzzy Set
IFWA	Intuitionistic Fuzzy Weight Aggregation
iGPS	Indoor Global Positioning System
IIL	Incapable Instrument List
IPP	Inspection Process Planning
IPPEX	Inspection Process Planning Expert
IPSD	Inspection Planning System Database
ISO	International Organization for Standardization
LAC	Local Accessibility Cone
LADAR	Laser Detection and Ranging
LICS	Local Instrument Configuration Space
LTL	Lower Specification Limit
LVC	Local Visibility Cone
LVM	Large Volume Metrology
LVM	Local Visibility Map
LVMIS	Large Volume Metrology Inspection System
MADM	Multi-attributes Decision-making
MC	Measurability Characteristics
MP	Measurement Plan

MPE	Maximum Permissible Error
MS	Measurable Space
NC	Numerically Controlled
NRK	New River Kinematics
OC	Oder Crossover
OFS	Occluded Face Set
OOPIPP	Object-oriented Planner for the Inspection of Prismatic Parts
PAD	Point Accessibility Domain
PLM	Product Lifecycle Management
QC	Quality Characteristics
SA	Spatial Analyzer
SCP	Set Covering Problem
SMR	Spherically Mounted Retroreflector
STL	Stereolithography
sTSP	Symmetric Travelling Salesperson Problem
SVM	Spherical Visibility Map
TOPSIS	Technique for Order Performance by Similarity to Ideal Solution
TRL	Technology Readiness Level
TSP	Travelling Salesperson Problem
UML	Unified Modelling Language
UTL	Upper Specification Limit
VC	Visibility Cone
VM	Visibility Map

LIST OF FIGURES

Figure 1.1 Inspection planning system hierarchy 2
Figure 1.2 Scope of the research 5
Figure 1.3 Overview of the thesis organisation 7
Figure 2.1 An inspection control hierarchy (Hopp and Lau, 1983) 10
Figure 2.2 Architecture of the inspection programming system (Hopp, 1984) 10
Figure 2.3 The feature-based inspection planning system (EIMaraghy and Gu, 1987) 11
Figure 2.4 Intelligent inspection planning environment (Yau and Menq, 1992)....... 13
Figure 2.5 The structure of the developed CATIP system (Limaiem and ElMaraghy, 1999) 15
Figure 2.6 Framework of the proposed work (Lin and Murugappan, 1998) 15
Figure 2.7 Flow chart of integrated applications of the developed system (Zhang et al., 2000)... 16
Figure 2.8 Layout of OOPIPP (Beg and Shunmugam, 2002)................................... 17
Figure 2.9 Overview of the proposed inspection planning strategy (Cho et al., 2004) 18
Figure 2.10 Dimension and tolerance specified on a part 26
Figure 2.11 Classes of tolerance (Albuquerque et al., 2000) 28
Figure 2.12 Tolerances classification (Mohib et al., 2009)....................................... 29
Figure 2.13 Methodology of the feature accessibility analysis (Ziemian and Medeiros, 1997). 33
Figure 2.14 Accessibility analysis system overview presented in Alvarez et al. (2008a; 2008b) 33
Figure 2.15 Examples of Gauss maps and visibility maps (Woo, 1994) 35
Figure 2.16 Obtained GAC of target point based on the CAD model (a) half-line probe (b) grown half-line probe (Spitz et al., 1999) 40
Figure 2.17 A VM for 12 surfaces of an object 46
Figure 2.18 Sandwich cutting process 46

Figure 3.1 The modelled interactions among inspection activities 54
Figure 3.2 Individual functions supported by the proposed system........................... 55
Figure 3.3 The use case representation of the framework 62
Figure 3.4 The proposed IPP for LVM system structure 63
Figure 4.1 GD&T taxonomy presented in UML class graph 67
Figure 4.2 Datum classification in UML class graph 68
Figure 4.3 Data structure of the IPSD 68
Figure 4.4 Structure of the tolerance analysis module in UML 70
Figure 4.5 Workflow of the tolerance analysis 71
Figure 4.6 An example of inspection tasks identification by tolerance analysis 72
Figure 4.7 The retrieve of the parameters of datum A 72
Figure 4.8 The retrieval of parameters of surface profile tolerance........................... 73
Figure 5.1 Trade-off relationship between measuring uncertainty and process tolerance (Flack and Hannaford, 2005)... 76
Figure 5.2 An example of decision rules for part inspection 80
Figure 5.3 An example of simple acceptance and rejection using a 4:1 ratio (ASME, 2001). 82
Figure 5.4 System structure of instrument selection in UML 85
Figure 5.5 UML activity diagram of instrument selection-Phase 1 88
Figure 5.6 UML activity diagram of instrument selection-Phase 2 96
Figure 5.7 UML activity diagram of fuzzy decision making process 97
Figure 5.8 Examples of points distribution on line, plane and lobed circle 106
Figure 6.1 Data structure of a binary STL file (Wikipedia).................................... 111
Figure 6.2 An example of convex set... 112
Figure 6.3 Illustration of convex hull... 112
Figure 6.4 An example of concave region 113
Figure 6.5 Illustration of half space 114
Figure 6.6 An exemplary VC of a point **p** on a facet *F* 115
Figure 6.7 An exemplary LVC... 116
Figure 6.8 Measurable space of a laser tracker 117
Figure 6.9 A 2D example of instrument configuration space 118
Figure 6.10 2D example of configuration space obstacle 119
Figure 6.11 View-frustum culling technique 121
Figure 6.12 Visibility analysis employing different visibility culling techniques ... 122

Figure 6.13 Hull facets and non-hull facets on a pocket 124
Figure 6.14 Example of identifying occluded faces 125
Figure 6.15 Examples of neighbouring occluders and potential occluders Liu et al. (2009) 126
Figure 6.16 Bounding volume examples 128
Figure 6.17 Different DOPs (Kimmerl, 2005) 128
Figure 6.18 More details are obtained as the number of bounding boxes increases 129
Figure 6.19 Compromised viewable space caused by different bounding volumes 130
Figure 6.20 A laser tracker simplified by three bounding boxes 130
Figure 6.21 The constructed convex hull of a model that consists of 192135 points (Hert and Schirra) 131
Figure 6.22 Comparison of non-viewable space due to obstacles 133
Figure 6.23 (a) Spherical projection of a triangulated hexagonal surface to a sphere 136
Figure 6.24 Example 1 of visibility analysis 138
Figure 6.25 Process of calculating the GICS 140
Figure 6.26 Example 2 of visibility analysis 142
Figure 6.27 Example 3 of visibility analysis 144
Figure 7.1 An instance of the minimum set covering problem 148
Figure 7.2 An example of clustering process using 2D GICSs 150
Figure 7.3 Set covering presentation of the example 151
Figure 7.4 Example of where non-optimal solution is obtained 156
Figure 7.5 The process of intersecting GICSs 162
Figure 8.1 GICSs of three inspection points 165
Figure 8.2 Discretization of the solution space 169
Figure 8.3 Discretization of the GICS using six planes 170
Figure 8.4 Simulation of the measurement uncertainty using SA 171
Figure 8.5 Simulated uncertainty information of an inspection point 171
Figure 8.6 Examine the part using different measuring sequence 172
Figure 8.7 An aTSP example with all solutions 175
Figure 8.8 Example of modelling inspection locations to TSP cities 179
Figure 8.9 Modelled TSP with safe bounding volume 180
Figure 8.10 The proposed approach to modify penetrating path 181

Figure 8.11 Inspection two sets of points by changing the direction of the photogrammetry system 182
Figure 8.12 Path planning without consideration of obstacles 183
Figure 8.13 Path planning using 3D Euclidean distance and geodesic distance...... 184
Figure 8.14 Typical processes of a genetic algorithm........ 186
Figure 8.15 Proposed genetic algorithm 187
Figure 8.16 Structure of chromosomes 188
Figure 8.17 Roulette wheel selection of chromosomes 190
Figure 8.18 Crossover process 191
Figure 8.19 The proposed genetic algorithm 194
Figure 8.20 Instrument locations........ 196
Figure 8.21 Sequencing result with pop_size of 30 and 1000 iteration 197
Figure 8.22 Sequencing result with pop_size of 100 and 1000 iteration 197
Figure 8.23 The solution history of the optimal sequence planning 198
Figure 8.24 Inspection points on the part........ 200
Figure 8.25 Imported locations of the inspection points........ 200
Figure 8.26 Generated 3D path for the inspection process 201
Figure 8.27 Approximate optimal inspection path........ 202
Figure 8.28 Solution history of the approximate optimal path 202
Figure 8.29 A multiple instruments configuration 204
Figure 8.30 Individual optimal paths for three instruments........ 205
Figure 8.31 Optimized inspection paths for three individual instruments 205
Figure 9.1 An example part with GD&T specification........ 208
Figure 9.2 Features on the example part 209
Figure 9.3 Points distribution on feature elements 216
Figure 9.4 Concave and convex region on the part........ 217
Figure 9.5 Global instrument configuration spaces for convex features 219
Figure 9.6 Global instrument configuration spaces for concave features 221
Figure 9.7 Distribution of all GICSs 222
Figure 9.8 Clustering analysis results showing the final clusters in two viewing directions 223
Figure 9.9 Discretized GICSs and sampling points 224
Figure 9.10 Optimal locations of the instrument in $GICS(S_1)$ and $GICS(S_6)$........ 226
Figure 9.11 TSP modelling and solution of S_1 227

Figure 9.12 TSP modelling and solution of S_6 229
Figure 9.13 The wing box with inspection points.. 230
Figure 9.14 Global instrument configuration spaces for convex features 237
Figure 9.15 Global instrument configuration spaces for concave surfaces 238
Figure 9.16 Distribution of all GICSs in different views.. 244
Figure 9.17 Clustering analysis results 247
Figure 9.18 Discretized GICSs and sampling points 240
Figure 9.19 Optimal locations of the laser tracker in $GICS(S_1)$ and $GICS(S_9)$ 242
Figure 9.20 TSP modelling and solution of S_1 249
Figure 9.21 TSP modelling and solution of S_9 250
Figure 9.22 Multiple TSP modelling and solution for two laser trackers 252

LIST OF TABLES

Table 2.1 Supported polygonal and CAD formats in Polyworks and SA (Innovmetric, 2012; NRK, 2012) 25
Table 2.2 Tolerance definitions in BS EN ISO 1101(2005) 27
Table 2.3 Knowledge-based clustering algorithm for inspection planning of CMMs (Ajmal and Zhang, 1998) 43
Table 2.4 Clustering results of the VM 46
Table 3.1 Research outcomes and deliverables with associated chapters 57
Table 4.1 Identified inspection tasks from tolerance 2 73
Table 5.1 Proposed TRL levels 84
Table 5.2 Example of crisp MCs 87
Table 5.3 Stored data of FARO Laser Tracker 87
Table 5.4 Linguistic performance and IFNs 91
Table 5.5 Linguistic importance and IFNs 91
Table 5.6 Instrument performance rating 91
Table 5.7 DMs importance and corresponding weights 99
Table 5.8 Performance rating of alternatives 99
Table 5.9 Assigned importance for all criteria 100
Table 5.10 Separation measure and relative closeness coefficient 102
Table 5.11 Assigned importance for all criteria-case 2 102
Table 5.12 Separation measure and relative closeness coefficient-case2 103
Table 5.13 Minimum number of points in BSI 7172 105
Table 8.1 Simulation of combined uncertainty to all inspection points 170
Table 8.2 The performance and the polynomial complexity of some algorithm 178
Table 8.3 Running results of the instrument location planning 195
Table 8.4 Running results of the measurement sequencing process 199
Table 9.1 Description of tolerances and dimension in the example part 209
Table 9.2 Description of datums in the example part 210
Table 9.3 Decomposed GD&T specification with respect to feature elements 210

Table 9.4Decomposed datums with respect to feature elements 210
Table 9.5 Inspection task list.. 211
Table 9.6 Quantitative requirements of the tasks.. 212
Table 9.7 Crisp MCs of the inspection tasks.. 212
Table 9.8 Available inspection systems with associated MCs................................ 213
Table 9.9 DMs importance and corresponding weights.. 214
Table 9.10 Performance rating of instruments.. 214
Table 9.11 Assigned importance for all fuzzy criteria.. 215
Table 9.12 Relative closeness coefficient and rank of individual instrument.......... 215
Table 9.13 Number of points on each feature element 216
Table 9.14 Simulated measurements of 18 inspection points on F_2 225
Table 9.15 Quantitative requirements of the tasks.. 231
Table 9.16 Crisp MCs of the inspection tasks... 231
Table 9.17 DMs importance and corresponding weights 232
Table 9.18 Performance rating of instruments.. 233
Table 9.19 Assigned importance for all fuzzy criteria.. 233
Table 9.20 Relative closeness coefficient and rank of individual instrument.......... 234
Table 9.21 Simulated measurements of 35 inspection points of T_4 at the optimal location.. 240

CHAPTER 1 INTRODUCTION

1.1 Research background

Inspection process planning (IPP) is considered as an important activity affecting both quality and cost in today's manufacturing industry. There is an increasing requirement of applying automated IPP for products with characteristics such as tight tolerance and high quality demanding optimized IPP to reduce inspection time and cost with minimum human interaction (Zhao et al., 2009). Moreover, in manufacturing and assembly industries such as aerospace and automotive, there is a constantly increasing requirement for integrating IPP within the early design stages in order to increase product performance and reduce costs by automation and elimination of rework (Maropoulos et al., 2008). Geometric Dimensioning and Tolerancing (GD&T) is deemed as one of the most important manners to specify engineering design and product requirements. Therefore inspection with respect to GD&T is broadly used to examine the conformity of manufactured products (Lee et al., 1997; Gao, 2006).

Intensive research effort with respect to IPP has been made over the past two decades for Coordinate Measuring Machines (CMMs) (Zhao et al., 2009) and a variety of different laser scanners (Li and Gu, 2004). Since CMMs are widely accepted due to 3D coordinates sampling capability with high accuracy, it is not surprising that most IPP researches were exclusively carried out for CMMs. Despite the different definitions and algorithms adopted in many proposed IPP systems, a common information flow is found stating the steps determining "what to measure" and "how-to-measure" based on the product data, shown in Figure 1.1 (EIMaraghy and Gu, 1987; Alting and Zhang, 1989; Feng, 1994; Yang et al., 1994). Activities including inspection task identification, inspection operation selection, sampling strategy, system configuration and task sequencing are taking place sequentially in order to generate an inspection plan that can be carried out by the specific instrument. The planning process begins with the function that identifies the inspection task according to the design information and product specification. Then suitable instrument and accessories are chosen to fulfil the various task requirements, for instance, tolerance level, cost and speed. Sampling strategy decides the number and location of discrete inspection points if a tactile sensor is used, or the pattern of scanning when a scanning probe is engaged. After configuring the inspection system in terms of placement and setup, the sequence of the inspection operations is derived with reduced inspection time.

Figure 1.1 Inspection planning system hierarchy

Large volume metrology (LVM) such as laser trackers, iGPS (indoor Global Positioning System), LADAR (Laser Detection and Ranging), photogrammetry and optical scanners (Puttock, 1978; Peggs et al., 2009) are rapidly advancing utilizing new technologies. Characterized with large volumetric coverage, high sampling speed and practical accuracy, LVM systems have replaced conventional inspection methodologies in industries such as aerospace and automotive, where large-scale products with high complexity are manufactured. In spite of constant development and emergence of new techniques, a recent literature review indicates the lack of available IPP methods for LVM systems. Although several attempts have been made to accomplish the measurability analysis and instrument selection (Cai et al., 2008; Cai et al., 2010; Muelaner et al., 2010), there is still a need for further work in this area to establish a systemic framework for LVM to define what to measure and how.

Digital manufacturing technology is increasingly employed in modern large-scale industries with the assistance of the latest 3-D visualization software, virtual reality, and product lifecycle management (PLM) systems (Roland et al., 1998; Waurzyniak, 2003). These enterprises benefit from this computer-based technique to influence product design using manufacturing feedback provided by virtual simulation. In the meantime, digital manufacturing provides comprehensive understanding of production issues during the development stage in order to evaluate the cost of manufacturing and eliminate engineering change orders. Benefiting from seamless integration with state of art computer-aided design (CAD) environment, modern metrology software embodies functions not only controlling the instruments and processing measurement data, but also fabricating and simulating inspection processes. This paves the way for taking the measurement and verification of large and complex products into account during early design within a digital manufacturing environment.

1.2 Research aim, methodology and scope

1.2.1 Research aim

Targeted at large-scale product inspection, the ultimate aim of the research is to develop an IPP system exclusively for LVM instruments deciding what characteristics should be measured and how. Moreover, the inspection planning should play a vital role in evaluating the design process.

As described previously, a complete IPP system should be capable of accomplishing tasks including inspection task identification, inspection operation selection, sampling strategy, system configuration and task sequencing. Therefore the presented system must embody individually functionalized modules to underpin the above inspection planning processes. A precise comprehension of the design is compulsory at the beginning of the process followed by the ability of specifying the inspection assignments and appropriate decision-making for suitable instruments with applicable inspection strategy. Optimized configuration of the instruments and effective sequence of the assignments are anticipated in order to achieve preferable inspection results with reduced cost. The system should be able to communicate with the selected instrument and manage the data flow in real time and finally generate inspection report according to the design requirements. The aim of objectives of this research are expanded and discussed in detail in Chapter 3.

1.2.2 Research methodology

The study carried out in this thesis aims to tackle an applied manufacturing problem and thus is considered as an engineering research activity. Research methods in the engineering category have been investigated for several decades (Walton, 1987; Trafford, 2001; Bryman, 2004). Generally a research process complies with either an inductive or deductive approach. The former collects data and then proposes a theory based on the data while the latter present the theory and prove it afterwards using data.

The deductive approach is adopted in this research to develop an innovative methodology ensuring the delineated aim is achieved. After identifying the background and research aims, a thorough literature review is conducted to outline the present IPP systems as well as the recent development of LVM systems. Having clearly specified the research gap with potential challenges based on the survey, a system is proposed to realize the demanded requirements. The validity of the system is examined by conducting case studies, which is the most effective way to verify the applicability of developed products to industrial application (Johnson and Johnson, 1991; Yin, 1993).

Unified Modelling Language (UML) is widely used as a standardised and object-oriented modelling language in the software industry. It consists of a variety of techniques that are applied in data modelling, business modelling, object modelling and component modelling (Mishra, 1997). The latest version, UML 2.4, contains 14 types of diagrams (OMG, 2013) to facilitate the modelling process. In this study, use case diagram and class diagram are used to present desired functionalities and data structure of the proposed system. The reader is recommended to refer to ISO/IEC 19505-1, 19505-2 (ISO, 2012) and BS ISO 19793 (BSI, 2008) for more information.

1.2.3 Research scope

The scope of the research is outlined in Figure 1.2. The main focus of the research is to develop a GD&T based inspection planning framework for LVM systems in a digital environment. Given the inherent difference of the measuring principles, the framework principally deals with single-point measurement instruments e.g. laser tracker, laser radar, articulated arm, iGPS and Photogrammetry. Instruments based on non-contacting digitising techniques were not incorporated in the study such as fringe-based scanners, time-of-flight based laser scanners and stereo vision scanners.

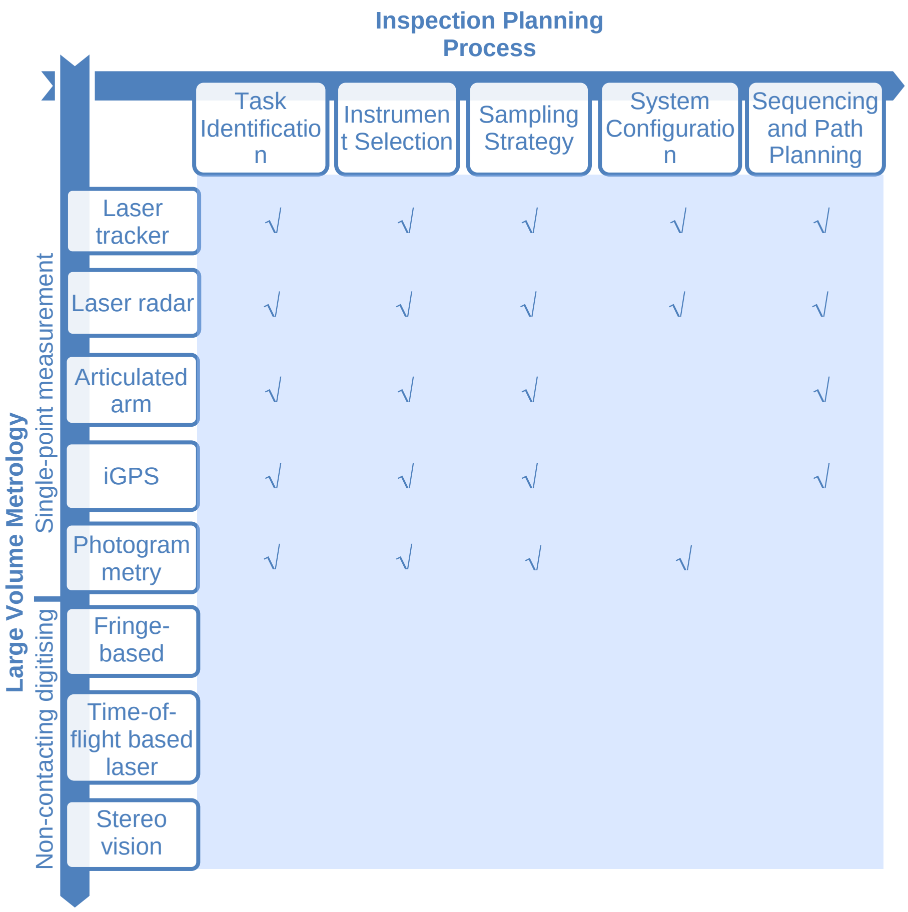


Figure 1.2 Scope of the research

The proposed system covers the inspection planning process including activities such as task identification, instrument selection, sampling strategy, system configuration and measurement sequencing and path planning. However, some of the functions were not applicable for certain instruments due to the unique constitution of the system e.g. the multiple joints of the articulated arm and the multiple transmitters of the iGPS. The details are given in Figure 1.2. An optimized inspection plan is created at the end of the process, guiding metrologists or the measurement operator to conduct the inspection. In addition, the functional modules can serve as extended functions to the metrology software carrying out individual activities such as visibility or line of sight check, instrument locating and uncertainty evaluation.

1.3 The structure of the thesis

The thesis is organized in 10 chapters shown in Figure 1.3. A general review of present computer-aided IPP systems primarily for CMMs and an introduction of LVM instruments with available commercial metrology software concerned in this research are given in Chapter 2. Research gaps are concluded after the review. In Chapter 3, research aims, objectives and stages are presented based on the expected functionalities of a typical IPP system. The proposed IPP system for LVM instruments is also described in this chapter with determined functional modules. This is followed by the details of each module presented from Chapter 4 to Chapter 8. The developed system is utilised to generate inspection plans for an artefact as well as a part from aerospace industry in Chapter 9. Finally, the conclusion is drawn in Chapter 10 including the specified contributions and limitations coupled with potential future research opportunities. The structure of the thesis is shown in Figure 1.3 indicating the contents of each chapter within the context of the research.

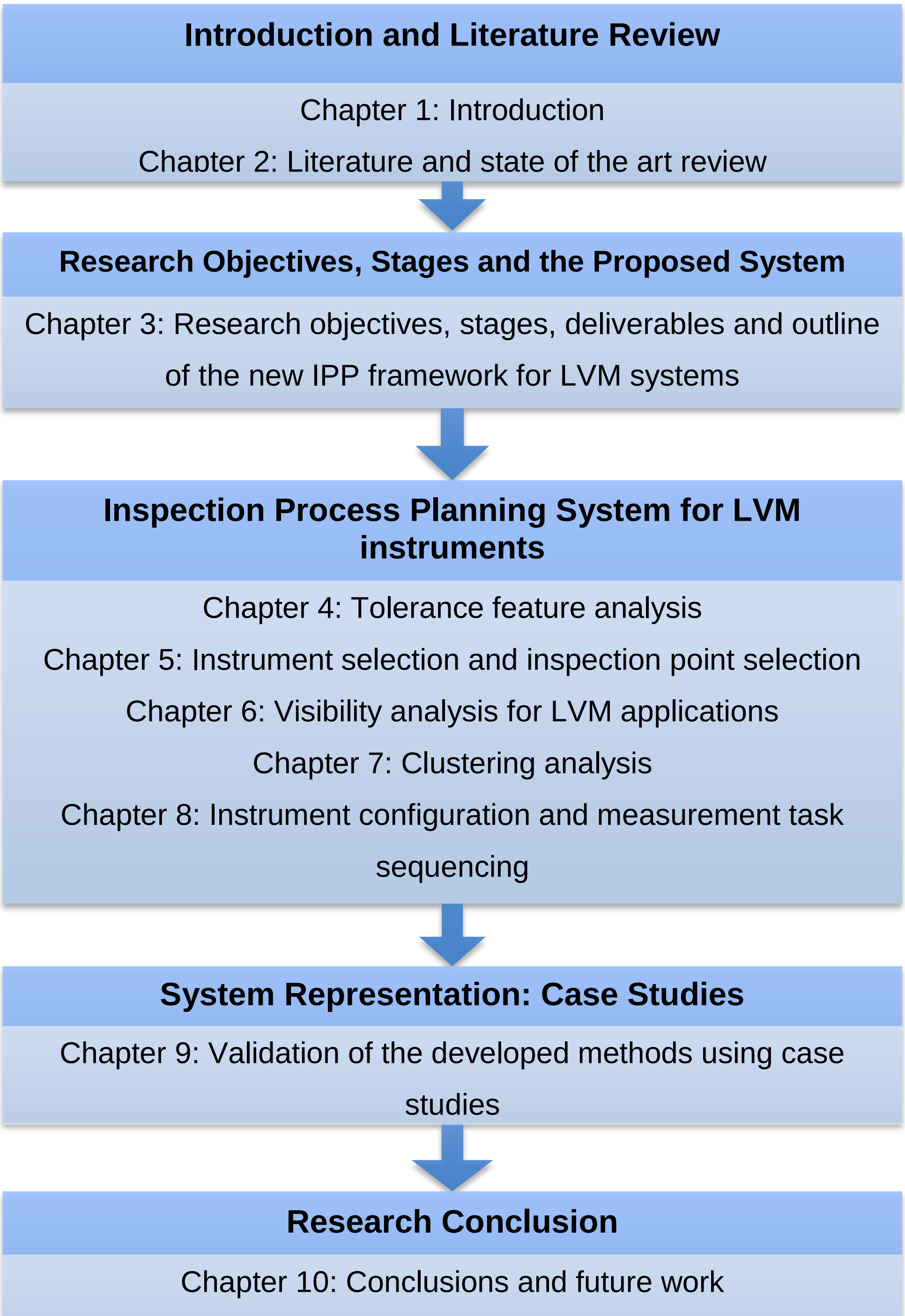


Figure 1.3 Overview of the thesis organisation

CHAPTER 2 LITERATURE AND STATE OF ART REVIEW

2.1 Introduction

As described in Chapter 1, inspection planning has been deemed as a critical activity that occurs between the design and the manufacturing process. Since IPP systems emerged in early 1980s, extensive research efforts were made to achieve appropriate and effective planning results. Nevertheless, CMMs dominate the product verification process due to its 3D coordinates sampling capability with high accuracy and therefore most research was exclusive to CMMs. In addition, this research aims to develop an IPP framework for single-point measurement systems that share similar processes in inspection planning with CMMs. This chapter thus focuses on the development of IPP research for CMMs in section 2.2 and recent developments of LVM systems in section 2.3. Commercial metrology tools are briefly reviewed in section 2.4 to identify their capabilities in terms of integrating with IPP system. State of art of Geometric Dimensioning and Tolerancing (GD&T) is given in section 2.5. Literature of visibility analysis is reviewed in section 2.6, followed by clustering analysis in section 2.7. Section 2.8 summaries the existing methods regarding measurement path planning for CMMs. Research gaps and challenges of proposing an IPP system for LVM instruments are finally highlighted in section 2.9.

2.2 Evolution of the IPP

Traditionally, inspection serves as a judgement that either accepts or rejects the product based on the measurement data and the designed tolerance. The process is a highly manual process and the results heavily depend on the experience of the metrologists. However, the role of inspection has recently changed to verify the manufacturing process in order to achieve quality objectives. This is due to the characteristics of modern manufacturing such as high value, variety production, tight tolerance and strict quality requirements (Maropoulos et al., 2008; Zhao et al., 2009). Under this circumstance, effective and accurate inspection is mandatory in order to meet the rigorous demands. Computer-aided design (CAD) and computer-aided manufacturing (CAM) are deemed as the two essential highly integrated technologies, which control design and manufacturing process utilizing computer resources via high performance hardware, algorithms, logic and the processing of information. Extensive developments of CAD/CAM have been conducted for automation and integration of various stages through the design and manufacturing process during the past two decades. Having realized the potentials brought by the computer based technologies, researchers started to facilitate the product verification process by using a computer-aided inspection planning system. By taking advantage of modern CAD and CAM techniques, not only can fast yet reliable decision be made, but also efficient and precise data exchange between the design and manufacturing processes can be executed.

2.2.1 Early development of IPP – feature-based approaches

Based on hierarchical and task-decomposition techniques, Hopp and Lau (1983) developed a control system for CMMs (Figure2.1). Their world model hierarchy aimed to decompose a number of high level planning objectives i.e. tolerance, probe selection and part setup into simpler goals such as path planning and surface fitting. Eventually, those tasks were decomposed down to the level of individual servo motor codes that commanded the CMMs.

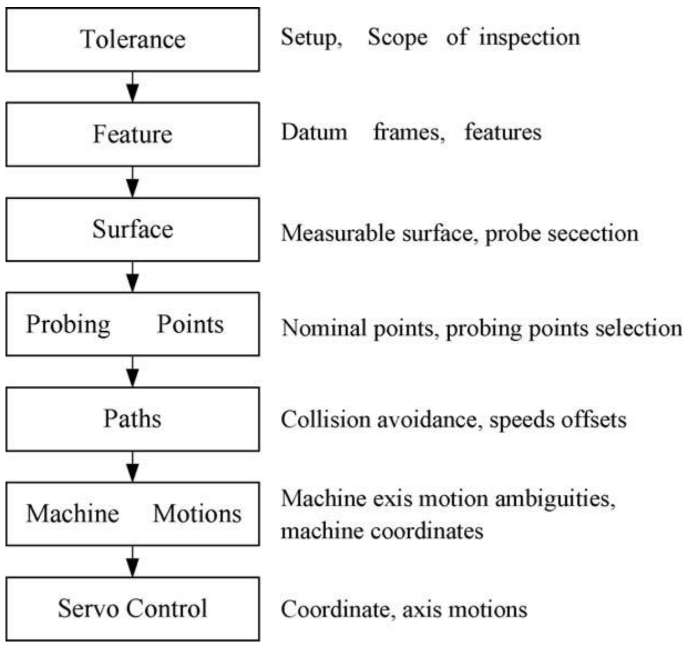


Figure 2.1 An inspection control hierarchy (Hopp and Lau, 1983)

Tolerancing information was specified by the user initially and the related features and datums were identified and transformed to mathematical description for derivation according to various standards. Nominal points were located on the target surface and datums for probing and a corresponding collision-free path was generated. The system was implemented to output a database that contained information regarding the decisions defining the goal decomposition at each stage. The database was then compiled and transferred into the control computer of the CMM to conduct the inspection. The structure of the implemented system is shown in Figure 2.2.

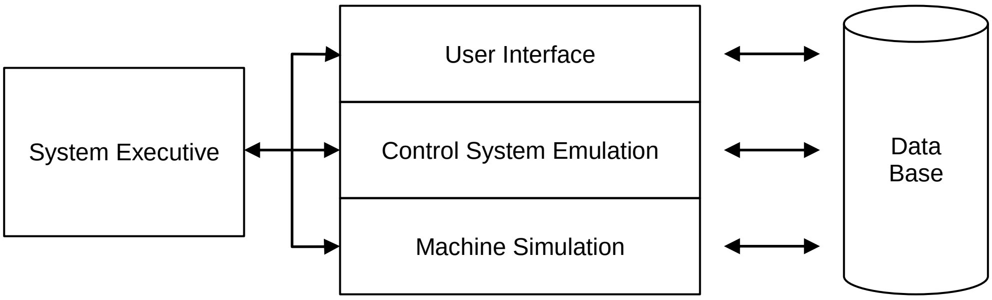


Figure 2.2 Architecture of the inspection programming system (Hopp, 1984)

EIMaraghy and Gu developed the first feature-based expert inspection planning task system for CMMs in 1987 (EIMaraghy and Gu, 1987). The research filled the gap of generative inspection planning by employing a knowledge-based expert system. Interpretation of engineering and drawings and activity planning were regarded as the most important steps throughout the process. Due to the lack of standardized data formats between commercial geometric modelling systems and process planning systems, the authors emphasized the linkage between the representation of the part and the process planning system by proposing a feature-orientated modelling and planning system in PROLOG. Three primary parts composed the system including: (1) feature base, (2) interactive modelling dimensioning and tolerancing module, and (3) knowledge base for tolerance assignment consultation. The feature base comprised information of the parts and the inherent relationship between the features and processes. Utilizing the integrated knowledge base, inference engine and knowledge acquisition interface, suggestions and recommendations were provided to the user interactively for assigning appropriate GD&T. The authors laid a number of knowledge rules based on the characteristics of CMM inspection to facilitate the processes including representing the features, selecting inspection facilities, analysing feature accessibilities and determining the inspection sequence. The structure of the system and the logical sequence of each module are presented in Figure 2.3. Although the system was only theoretical, it was the first attempt to integrate IPP with CAD environment.

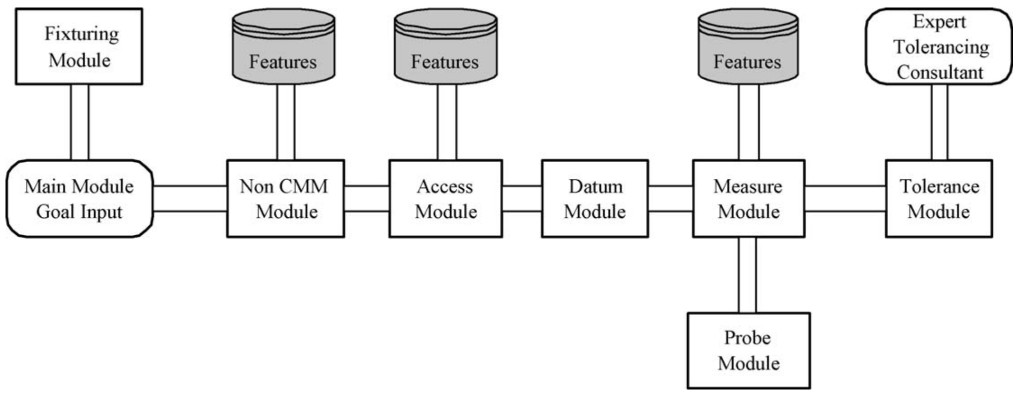


Figure 2.3 The feature-based inspection planning system (EIMaraghy and Gu, 1987)

Medland et al. (1990; 1993) claimed that the variations in products and the variety of CMMs limited the efficiency of inspection in productive environment. Research efforts were thus made to integrate CMMs with CAD systems, underpinned by effective communications and file exchanges. The developed demonstrator system identified features to be measured together with tolerances and generated probing geometry and approach vectors. The uncertainty of the probe was taken into account and systematic treatment was employed to maintain the validity of the inspection such as increasing the number of inspection points that reduced the probing errors. Probing strategy was optimized according to criteria such as highest priority, minimum changes in orientation and shortest path and the generated plan was then transformed into codes that controlled the CMM to carry out the inspection. A data feedback module was created for the user to interrupt and enhance the operations based on the data transferred back from CMM. Improvement of design and elimination of manufacturing errors could thus be achieved.

Brown and Gyorog (1990) designed a prototype system to generate dimensional inspection plan using a generative process planning expert named IPPEX (Inspection Process Planning Expert). Product was modelled using a solid geometric modeller with a separate dimension and tolerance modeller specifying the design requirements. A database was integrated with the IPPEX containing information such as available machines, probes, and fixtures and inspection process knowledge was stored in the forms of production rules. Decisions were made through the inference engine in the system and eventually an inspection plan was outputted.

Aiming to reduce the cost induced by the process of understanding a drawing and creating the inspection strategy, Merat et al. (1991) developed an automated inspection planning system for CMMs within the Rapid Design Project. GD&T information was presented with features and each GD&T feature class was related to the Inspection Plan Fragment that contained inspection procedures such as sampled points and probe orientations. For instance, the diameter of a hole was corresponded to an Inspection Plan Fragment such that the probe of the CMM was inserted into the hole along the axis and a minimum number of three points must be measured. After generating all fragments, the system clustered all inspection request based on part orientation, and the inspection strategy that minimized the probe changes was chosen to conduct each tolerance feature. A collision-free path was created using a minimum path algorithm for 3D polyhedral objects in configuration space.

A five-module automated dimensional inspection environment was developed by Yau and Menq (1991; 1992a; 1992b) including: specification module, planning module, verification module, execution module and comparative analysis module (Figure 2.4). Centred at a knowledge-based inspection planner that assisted process flow and decision-making, inspection specification was gathered by translating functional requirements, tolerance, manufacturing attributes and CMM parameters. The results were then input into the planning module to determine sampling number and locations, such that the part could be represented completely and accurately. Probing vectors were decided afterwards based on the sampling size and dimensions of the part. The generated inspection path was validated by the verification module in the CAD environment ensuring a collision-free inspection. Finally, the execution module carried out the inspection plan, and obtained measurement data were returned to comparative analysis module to finalize the inspection report.

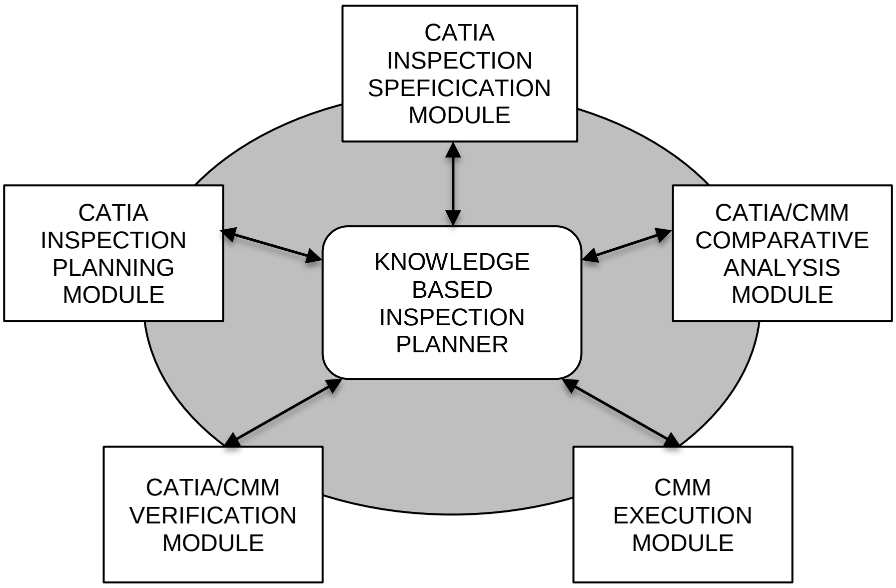


Figure 2.4 Intelligent inspection planning environment (Yau and Menq, 1992)

Originating from commercial quality requirements, Tannock et al. (1993) proposed a flexible and low-cost intelligent inspection planning and computer aided inspection system as assistance to the inspector and manufacturing or quality engineer. Predefined process plans with the part drawings were firstly input into the system using feature-based CAD representation to initiate the planning process. There were three types of inspection feature concerned in the research: (1) single geometrical dimension, (2) compound entities, and (3) geometrical tolerances. Based on the parameters associated with each type of inspection feature such as nominal values, size tolerances, measuring datums and state-transition preconditions, plan-reasoning process was conducted using a Stanford Research Institute problem solver. A sequence of inspection actions was thus identified and guidance for inspectors was given by the system.

2.2.2 Recent development of IPP- integration with CAD environment

Most IPP systems prior to 1995 were theoretical and the applications were heavily limited. ElMaraghy (1993) claimed that in order to maximize the potential of computer-aided inspection planning, the process must be integrated with CAD environment, which would benefit better utilization of resources, improved quality control, flexibility and enhanced logic formalization. In 1999, Limaiem and ElMaraghy (1999) developed a computer-aided tactile inspection planning system that featured four modules, specifically accessibility analysis, clustering module, sequencing module and path planning module (Figure 2.5). It incorporated a knowledge-based system to determine the number of location of inspection points according to the statistical distribution of the accuracy of the manufacturing process and tolerance information. By analysing the solid model of the part and surrounding objects, located inspection points were associated with accessibility information by using the intersection of concentric spherical shells with abstracted probe. Clustering and sequencing techniques were subsequently employed with precedence constraints to generate the sequence of measurement and resources allocation such as set-ups and probe orientations. A shortest path among all inspection points was generated eventually without interfering the part and surroundings.

Fan and Leu (1998) also developed an integrated solution for inspection planning for CMMs. By taking advantage of CAD techniques, automatic generation of measuring points was realized by applying some general rules of sampling basic geometrical features. In addition, the system allowed the user to specify any additional points on the surface to obtain better sampling results. The probe path was created by linking all safe points and measuring points using normal direction of the surface to avoid potential collision between the probe and the object. However, the above rules could not eliminate all collisions and therefore the authors applied a swept-volume analysis to detect any collision, which was treated by modifying the probe direction or inserting intermediate points for the probe path.

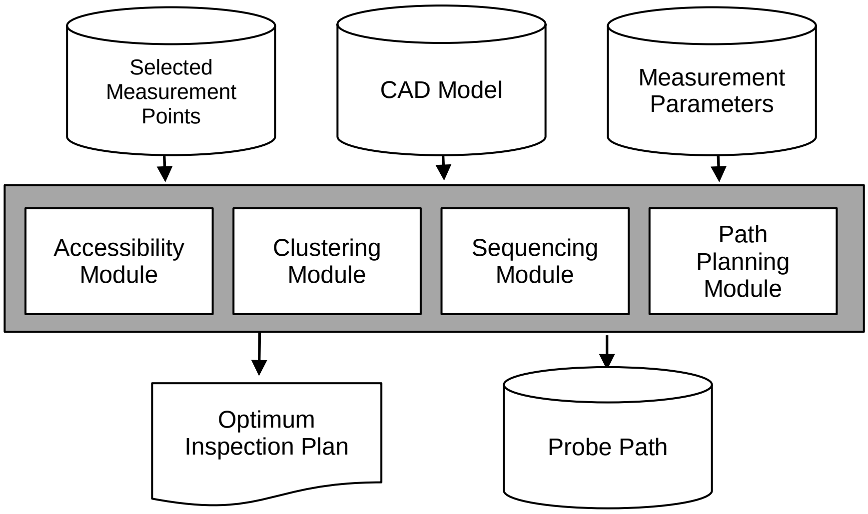


Figure 2.5 The structure of the developed CATIP system (Limaiem and ElMaraghy, 1999)

Lin and Murugappan (1998) proposed a framework that integrated the CMM into CAD/CAM environment such that the manufacturing and inspection activities were carried out simultaneously (Figure 2.6). Not concerning the inspection task identification, the research concentrated on the algorithm of detecting collision between the probe and the part. The user was asked to specify the starting and ending point of the probe and a modified ray tracing technique was adopted to check the interference. The geometries of the probe and the fixtures were ignored for simplicity.

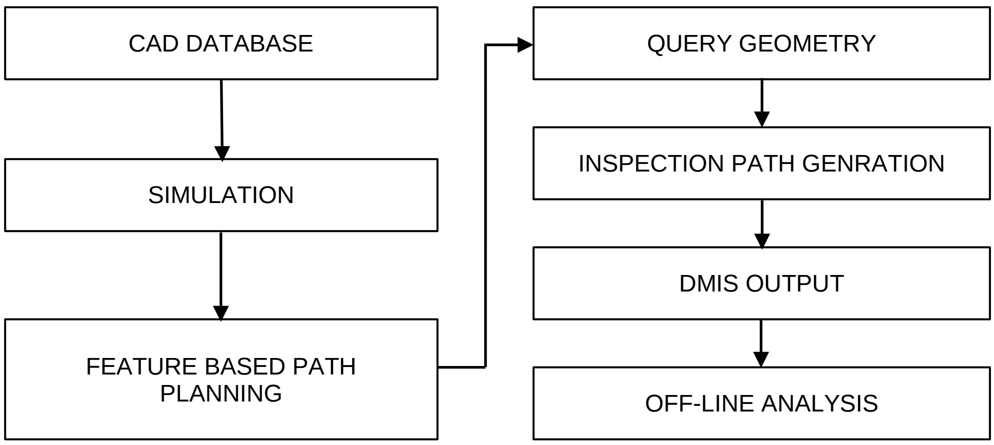


Figure 2.6 Framework of the proposed work (Lin and Murugappan, 1998)

Zhang et al. (2000) created a five-stage inspection planning system dedicated for CMMs (Figure 2.7). The tolerance feature analysis module enabled the user to select tolerance inspected by the CMMs with installed decision rules. The inspection feature was represented with a frame-based data structure that incorporated the tolerance information such as tolerance types and corresponding relationship to the features. The accessibility analysis module firstly represented each inspection feature using Gauss mapping, where the normal directions were obtained. An accessibility cone was then calculated as the dual image of the Gauss map, and discretisation of the accessibility cone was carried out to identify the accessible directions of the probe. One limitation of this research was that only six directions were considered as the possible orientations of the probe. A knowledge-based clustering algorithm subsequently grouped inspection features into feature families that shared the same probe orientation. The sampling number and distribution of the points were taken into account in the path generation module along with the determination of the measurement sequence. Offset points were inserted into the path of the probe to avoid collision and the therefore the trajectory of the probe was divided into the measuring movement and the approach movement. Finally, the generated plan was tested and simulated in a graphic environment to verify the effectiveness.

Figure 2.7 Flow chart of integrated applications of the developed system (Zhang et al., 2000)

Beg and Shunmugam (2002) presented an object-oriented planner for the inspection of prismatic parts (OOPIPP) in 2002 (Figure 2.8). The work incorporated an interactive collection of design data methodology to recognize the inspection features followed by a fuzzy logic selection of stable part orientation. The number of inspection points and the corresponding distribution were decided according to defined rules although only simple tolerance types were considered. Feature accessibility analysis was carried out by means of generating the accessibility cones, and the sequence of probe orientation and the measuring surfaces were determined in a way such that minimum inspection time could be achieved.

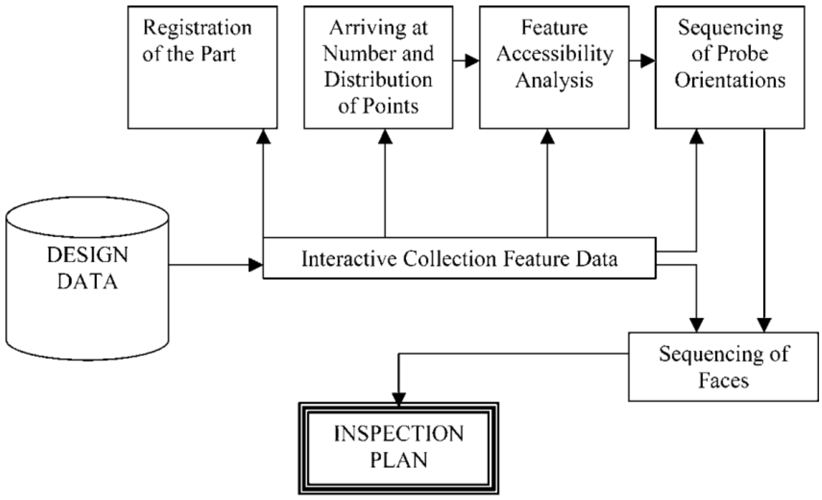


Figure 2.8 Layout of OOPIPP (Beg and Shunmugam, 2002)

Hwang et al. (2004) proposed a methodology that could minimize the number of part setups and probe orientations by applying a continuous Hopfield neural network. However, the inspection planning was treated as a mathematical problem rather than a practical manufacturing process and therefore the application was limited.

Cho et al. (2004) divided the inspection planning into two stages, namely global inspection planning that mainly focused on the feature level, and local inspection planning that concerned the inspection process on each feature of a part (Figure 2.9). Feature precedence tree was used to plan the inspection setup for a prismatic part on a machining centre. Each part was decomposed into a number of manufacturing features with information such as the probe approach direction representing the accessible directions of the probe to inspect the feature. After extracted from a CAD database and labelled with a predefined number, all features were grouped into feature groups based on knowledge rules to maximize the efficiency, and inspection sequence was obtained within each group by analysing the probe approach direction of each feature. Further decomposition of feature was carried out to generate inspection tasks and a hybrid neuro-fuzzy approach was used to determine the number of inspection points as well as their distribution. Collision was avoided by applying a specifically designed method, namely Z-map, and the traveling salesperson algorithm was implemented to create an optimal measuring path.

Apart from IPP for CMMs, there is very little literature available regarding inspection planning system for other instruments. ElMaraghy and Yang (2003), Mohib et al. (2009) and Fernandez (2008) proposed different inspection planning systems for laser scanner. However, the laser scanner concerned in those studies was only the scanner head that replaced the probe. Consequently, the systems were very similar to the IPP systems for CMMs. Other research concerning optical measurement instruments concentrated on the application of the technique and the accuracy performance and no complete IPP system was found (Li and Gu, 2004; Mohib et al., 2009; Zhao et al., 2009).

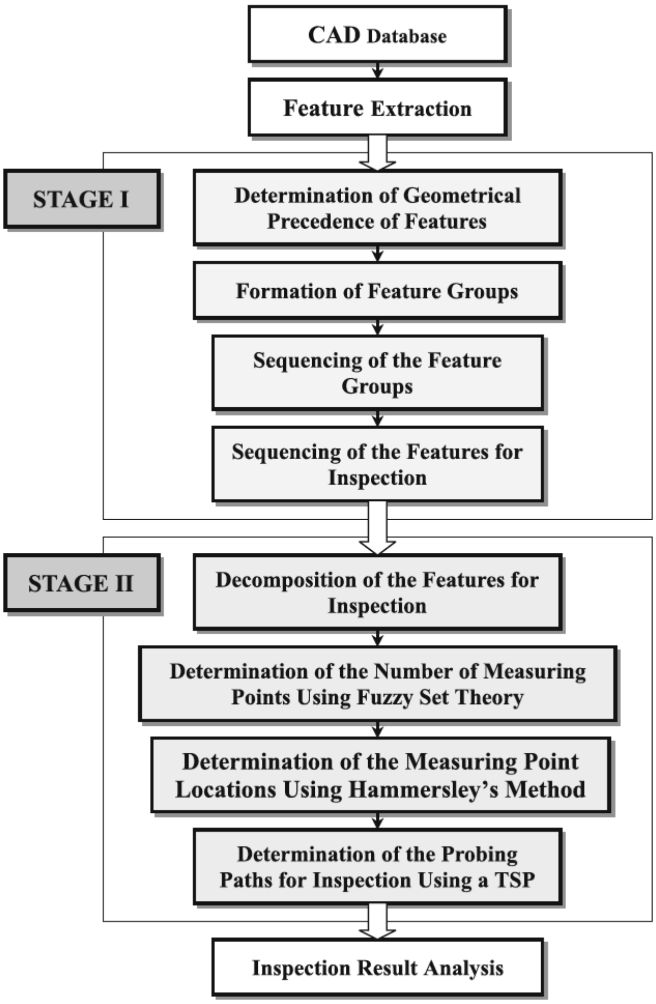


Figure 2.9 Overview of the proposed inspection planning strategy (Cho et al., 2004)

2.3 Development of LVM

Puttock (1978) defined the term “Large-scale metrology” in his review as “the metrology of large machines and structures”. The boundaries of this field are laboratory measurements at one end and surveying at the other. Neither boundary is well defined and will generally be confined to the metrology of objects in which the linear dimensions range from ten to hundreds of meters. He claimed that the length measurement was secondary challenge for the metrologist compared with the alignment and measurement of angle. Large-scale metrology was deemed as a significant challenge to the metrologist as the variation in objects was increasing as well as the higher requirement of accuracy. Therefore proper instrumentation and techniques for specific measurement task are required and a considerable degree of ingenuity is demanded as an attribute of the metrologist undertaking the work (Puttock, 1978).

There have been considerable developments and refinements in this field over the past 30 years with respect to technologies and instruments such as laser-based instruments, structured light and powerful digital photogrammetric methods. Large and complex assemblies and fabrications with complex surfaces are increasingly employed in the aerospace, power generation and automotive industries. Such industries strive to increase the quality specifications of large structures in order to improve product performance and reduce costs and assembly cycle times by automation and the elimination of rework during final assembly. These industrial requirements can be met by new technologies for large volume metrology (LVM) that are rapidly advancing due to the availability of high-speed electronics and inexpensive computer power (Maropoulos et al. 2008).

2.3.1 Laser tracker

Lau et al. (1986) first proposed the tracking laser interferometer known as the laser tracker in order to accomplish rapid and high-accuracy measurements conducted by industrial robots. There have been substantial improvements of this approach and applied to a variety of applications broadly rather than just to robot measurement. Commercial systems have been released by several companies with extended functions and corresponding assisted software. Estler et al.(2002) reviewed typical applications in the aerospace, automotive, and shipbuilding industries including:

1) workpiece inspection, including large antenna dishes, ship propellers, and aerofoils;
2) layout and fabrication of large assembly tools in aircraft manufacture;
3) reverse engineering in the auto industry, for example, a clay model can be digitized and the data used to mill a die for sheet metal stamping. The same tracker can measure the die and the stamped part;
4) volumetric accuracy checking of large machine tools and CMMs, including hexapods;
5) relative positioning for large part assembly.

Generally, a spherically mounted retroreflector (SMR) is used as the target and several types of retroreflector have been developed such as hollow cube corner SMR, glass cube corner SMR, and cat's eye reflector.

The laser tracker is deemed as a portable, frameless coordinate measurement system measuring the position of target in a spherical coordinates but is constrained by the line of sight, which requires a continuous laser beam, connected between the target and the home station. If the beam is obstructed or there is excessive acceleration of the target, the radial coordinate is required to reset at home position or measured reset points. In order to cope with the constraints caused by line of sight, several new approaches were proposed and demonstrated including absolute distance measurement (ADM) and integration with track arm.

2.3.2 Coherent laser radar

Although the laser tracker has realised more functions and facilitated coordinate measurement significantly, the use of cooperative targets such as cube corner retroreflectors induces several constraints such as measurement speed, labour requirements, and time consumption, which increase cost considerably. A new ranging technology named coherent laser radar emerged in the large volume metrology region that is able to overcome limitations.

A laser beam is targeted on the measured point or surface that is linearly modulated in time, and the reflected beam is received using the same optical system. As the frequency shift is proportional to the round trip transit time of the beam, the distance is calculated by identifying the difference between the received signal and the reference signal. The ranging system is served by a two-axis beam steering system with angle encoders. As there is no cooperative target, the system is capable of measuring an array of precise range measurements on the surface and then generate a three dimensional image of the measured surface by means of specific software (Menon et al., 2001). This technique is preferred if inaccessible surfaces are applied which cannot locate a SMR such as the interior of a toroidal fusion reactor (Kugel et al., 2001). Comparison between CAD presentation and the coordinate data can be generated in real time for alignment or assembly operations, which is vital for inspection of quality. Additionally, probe error is eliminated as measurements are taken directly from the object (Kugel et al., 2001).

The commercial laser radar is mainly supplied by Nikon. The company claims that the product is capable of a variety of quality assurance applications including part-to-CAD comparison and first article inspections, incoming quality assurance, in-process quality assurance, and outgoing quality assurance. Some key features are listed (Nikon, 2011):

1) High measurement throughput
2) Lower resource costs
    a. One-man operation
    b. Reduced fixed tooling requirements
3) Portable system enables on-site measurements
4) Minimum setup time
5) High accuracy measurements
6) Enables difficult or previously impossible metrology jobs

2.3.3 iGPS

Indoor GPS (iGPS) is a newly developed laser based measuring system for large-scale metrology. The relative portability, reconfigurability and ease of installation make the iGPS suitable for many industries manufacturing large scale products (Maisano et al., 2008). The system is utilised for parts inspection and reverse engineering, tracking and robotic control, shipbuilding, large-scale metrology, and large-scale assembly and alignment. The system consists of a number of laser transmitters that emit infrared light signals throughout the work volume and the signals are received by various types of sensors and then sent to the receivers. Angle data are generated by the receivers according to the original signal and specific software processes the angle data into highly accurate position and orientation information and makes this information available to all users. The main supplier of iGPS, Nikon, claims that the accuracy level is approaching laser tracker in areas 10×10 meters and is competitive with laser trackers for larger regions (Nikon, 2011). iGPS also features flexible configuration of the system depending on the requirements and instantaneous beam reacquisition, which helps the user significantly.

2.3.4 Photogrammetry

Photogrammetry was initially used in 1960s for the calibration of the reflector surfaces of large radio telescopes, parabolic and spaceborne deployable antennas, and shipbuilding measurements. Aircraft and aerospace manufacturing started to employ this approach in the early 1980s for the dimensional inspection of tooling fixtures, assemblies and master gauges (Estler et al., 2002).

This measurement technique is able to supply the shape, size, and position of objects by capturing a number of two dimensional images at different positions and the three dimensional coordinates of measured points are calculated by optical triangulation from these images. With the improvement of electronics such as computers and CCD cameras, photogrammetry has developed rapidly in the past two decades and formed the digital close-range photogrammetry known as vision metrology rather than filmed based close-range (Mikhail et al., 2001).

There are two main categories of industrial photogrammetric systems: on-line and off-line. Real-time or near real-time measurements can be achieved by using online systems composed of multiple digital cameras. By contrast, the images provided by single-camera systems can be processed in the specific software and generate off-line results of the measurements. The latest advances in camera technology drive the development of the term ‘on-line photogrammetry’. Integration of camera, optical sensor, microprocessor and communications technology enables the real-time process of the captured images within the camera. In other words, this smart camera is capable of acquiring an image, measuring any suitable artificial targets automatically, and downloading only image coordinates to a computer. In additional to the smart camera, potable and handheld touch probes are optional for interactive on-line measurement instead of conventional target. The probe assists the user to obtain access to places, which are not easily targeted such as edges, holes, and surfaces, as well as the features out of the camera’s view (Estler et al., 2002).

Although there are other available instruments at the moment but the above four categories are the most applied, together with additional equipment and accessories such as laser arm and T-mac.

2.4 Metrology software

With the increasingly developing metrology market, metrology software is advancing rapidly today. Most instruments are accompanied with adequate software for the user to control the device and collect and analyse the acquired data. However, modern manufacturing environment normally contains more than one particular instrument and many assignments must be accomplished by multiple instruments simultaneously. Consequently, there is a requirement of a general platform that can interface with all instruments and control and collect data from them.

PolyWorks (Innovmetric, 2012) and Spatial Analyzer (NRK, 2012) are the two major software solutions for 3D dimensional control as well as reverse engineering at present. Both of the platforms are able to incorporate instruments such as laser trackers, articulated arms, photogrammetry systems and hand-held probing devices. A 3D graphical environment is normally supplied to the user and simultaneously communication between the software and the instruments are supported. Real-time inspection is carried out and related data is collected by the system automatically. Additionally, the software is capable of implementing spatial transformation, fitting of nominal data, comparison of geometry, simulation of uncertainty, and integration of multiple instruments. Dynamic report is another feature that enables the user to produce conformance or uncertainty reports during the inspection. In order to assist modern automated inspection process, the software also supplies the ability of comprehending user-programmed codes and implementing the commands to accomplish fully automated inspection.

However, there are some differences in the two software products. Polyworks was firstly designed for high-density point cloud 3D digitising platforms. It thus excels in processing point cloud data and editing meshes polygonized from the cloud. It supports functions such as virtually assembly of the scanned product components, identification of the interfering parts and highlights instability in the assembly process (Innovmetric, 2012). In addition, activities in reverse engineering including identification and extraction of CAD entities on scanner models, modification and refinement of the models and finite element meshing are also supported with advanced algorithms and data management (Innovmetric, 2012).

In contrast, SA is specialised in supporting portable large-scale single-point metrology systems. Measurement uncertainty is taken into account for every measurement and the user can improve the accuracy using an ideal network by integrating multiple instruments into one measurement system (NRK, 2012). For instance, it is required to deploy multiple instruments in some scenarios such as aeroplane assembly and shipbuilding and this function enables the user to consider the uncertainties from all live instruments with specific uncertainty mathematic models. GD&T information embedded in a CAD model or primitive geometry can be extracted to SA and GD&T inspection can then be carried out. Measurement planning is integrated within the software so the user can generate pre-defined plans and simulate it subsequently to evaluate the uncertainty. A software development kit (SDK) is available to the user and VB script can be used directly to call all functions. This paves the way for automating both the planning and the conducting of the measurement process.

The supported polygonal and CAD formats for Polyworks and SA are given in Table 2.1.

Table 2.1 Supported polygonal and CAD formats in Polyworks and SA (Innovmetric, 2012; NRK, 2012)

	Polyworks	SA
Supported Polygonal	DXF, Inventor, Nastran, OBJ, PLY, POL, STL, VRML2.0	STL, Inventor
Supported CAD	CATIA V4 and V5, IGES, Inventor, JT, PRO/E, Solidworks, STEP, UG	CATIA V4 and V5, IGES, Inventor, PRO/E, Solidworks, Creo Parametric

2.5 Geometric dimensioning and tolerancing

Geometric dimensioning and tolerancing is deemed as one of the most essential procedures throughout the product lifecycle. Not only do process engineers determine the appropriate process for manufacturing the product, but also quality engineers rely on this tool to verify the product with respect to the designed specification (Ge et al., 1992). This is especially evident for a process such as assembly, where the variations of joining parts are crucial. As implied by the terms, GD&T contains two aspects (BS EN ISO 1101, 2005):

1) Dimensioning: defines the nominal geometry as designed in the drawing or CAD model.

2) Tolerancing: defines the permitted variation of the actual feature from the nominal value or the permitted variation among a number of features.

The example shown in Figure 2.10 indicates the two parts of the GD&T. The position of the hole on the part is interpreted by both the basic dimensions and the tolerance information, which is presented with respect to the two datums.

2.5.1 GD&T standards

As far as the standards are concerned, there are primary two widely accepted standards, namely the American Society of Mechanical Engineers (ASME) Y14.5-2009 and the Geometrical Product Specification (GPS) ISO/TR 14638: 1995. Despite the competitive relationship in present industrial applications, ASME standard focuses on describing the comprehensive specification of the design while GPS considers the entire production process from design to verification (Cristofolini et al., 2001). The difference and similarity between the two standards are revealed in a comparative study by Cristofolini et al. in 2001. This study is carried out based on the latest British Standard BS EN ISO 1101:2005 – Geometrical Product Specifications (GPS) - Geometrical tolerancing, which is identical with ISO 1101:2004.

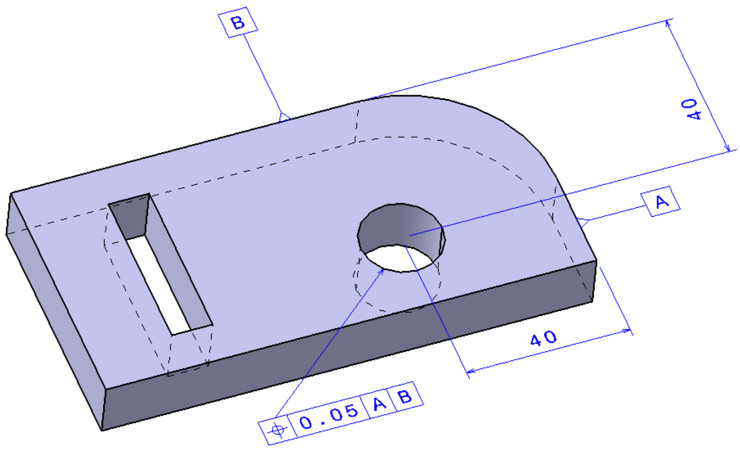


Figure 2.10 Dimension and tolerance specified on a part

2.5.2 GD&T classification

According to BS EN ISO 1101:2005, the tolerance can be divided into four categories: form tolerance, orientation tolerance, location tolerance and run-out tolerance. A total number of nineteen characteristics are grouped into those four categories based on their geometric definitions. The representing symbols and datum requirements are shown in Table 2.2 (BS EN ISO 1101:2005). In the few studies that attempted to link inspection task with GD&T information, several classifications of tolerances were proposed. Ge et al. (1992) applied the geometric classification as mentioned above to identify the corresponding geometric entities.

Table 2.2 Tolerance definitions in BS EN ISO 1101(2005)

Tolerances	Characteristics	Symbol	Datum needed
Form	Straightness	—	no
	Flatness	⏥	no
	Roundness	○	no
	Cylindricity	⌭	no
	Profile any line	⌒	no
	Profile any surface	⌓	no
Orientation	Parallelism	//	yes
	Perpendicularity	⊥	yes
	Angularity	∠	yes
	Profile any line	⌒	yes
	Profile any surface	⌓	yes
Location	Position	⌖	yes or no
	Concentricity (centre points)	◎	yes
	Coaxiality (axes)	◎	yes
	Symmetry	⌯	yes
	Profile any line	⌒	yes
	Profile any surface	⌓	yes
Run-out	Circular run-out	↗	yes
	Total run-out	⌰	yes

The study concentrated on the linkage of GD&T information with computer entity modeller and the selection of inspection task was not covered. Albuquerque et al. (2000) classified the tolerances into feature tolerances and face tolerances shown in Figure 2.11. A surface list was generated eventually by analysing the relations in feature tolerances and corresponding datums. The GD&T data was combined with feature types in the research carried out by Beg and Shunmugam (2002). They claimed that the inspection planning system should incorporate GD&T extraction in order to integrate the design and inspection. However, instead of developing a system, their approach relied on user interaction by asking a number of predefined questions. Shown in Figure 2.12, Mohib et al. (2009) used a classification based on ISO and AMSE Y14 standards to identify inspection features as well as inspection decisions. By constructing a decision matrix indicating the relationships among geometric features, manufacturing features and inspection features, knowledge rules and adjustable parameters were employed to identify inspection tasks. Nevertheless, the proposed method was not automated and only applicable when expert interaction was inputted.

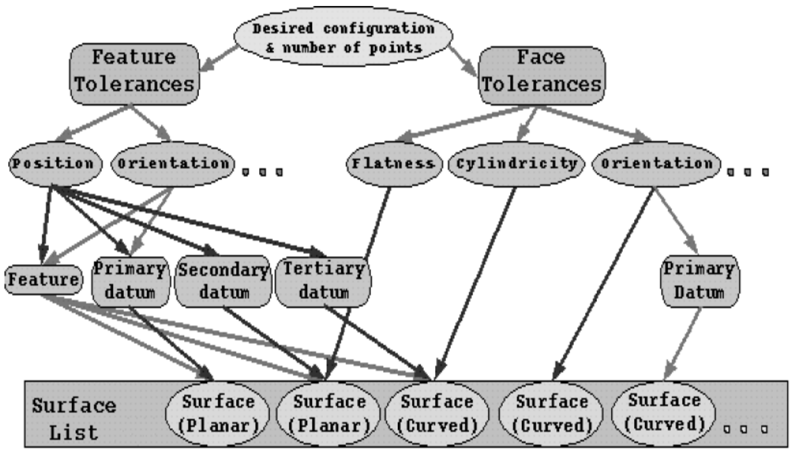


Figure 2.11 Classes of tolerance (Albuquerque et al., 2000)

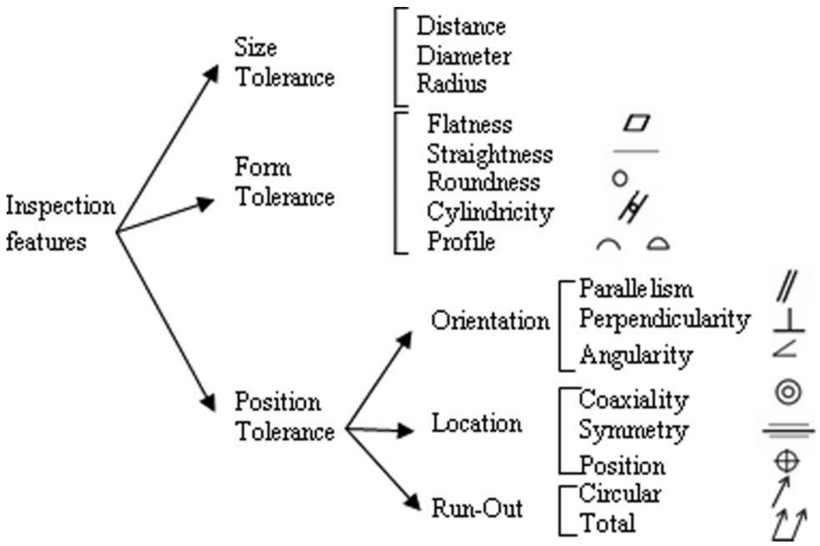


Figure 2.12 Tolerances classification (Mohib et al., 2009)

2.6 Visibility analysis

Visibility is an intensively defined and studied concept that exists in various domains e.g. computer graphics, computer vision, robotics and computational geometry, according to the multidisciplinary survey by (Durand, 2000). In spite of the different terminologies and methodologies in those disciplines, visibility analysis has attracted a vast amount of research interest for the last two decades, striving to assist manufacturing processes such as mould design, numerically controlled (NC) machining, assembly planning and CMMs inspection design.

2.6.1 Systems that compute approximate solution

In computer graphics, hidden surface/line removal is a widely applied technique, which determines all objects in the scene e.g. parts, polygons and lines that are visible along a given direction (Durand, 2000). Visible set problem, by contrast, is the synonym for the identification of all visible directions of a given object (Liu et al., 2009).

Spyridi and Requicha (1990) first considered determining the orientation of probes as accessibility analysis process for automatic inspection of CMMs. Accessible direction of an inspection point, known as accessibility cone (AC), was defined and computed with respect to a semi-infinite line, which represents the abstracted straight probe. Local accessibility cone (LAC) and global accessibility cone (GAC) were identified respectively by taking into account the immediate neighbourhood of a point or the entire workpiece. Involving methods such as Gaussian images and Minkowski sums, computation of LAC and GAC was carried out for polyhedral objects and simple geometric domains. Clustering process was then implemented based on the accessibility analysis results with the objective of finding the maximum intersection of GACs for all inspection points and features. Several suboptimal solutions were finally obtained as the output of the system since the minimal clustering problem is NP-complete.

Lim and Menq (1994) followed Spyridi and Requicha's AC definition but applied ray tracing method computing the intersection instead of Minkowski sums to save the expensive computational power. Three simplifications were made to facilitate the process including touch probe, feature path and accessibility cone. The probe and accessory were abstracted by a number of infinite lines, which took the volume of the stylus into account. Having considered discrete points rather than entire feature as the inspection target, occurred null cone could be treated as inaccessible point and removed in advance of probe path planning whereas computer memory was compromised. As far as probe orientation was concerned, information contained in AC such as distance of collision and possible orientation of the probe while not engaged were ignored in their research and therefore, 3D AC was transformed into 2D accessibility map which reduced the searching time and eliminated the attribute of target position. Modified ray tracing method was implemented for 720 possible angles of the PH9A probe head and LAC and GAC were created based on the generate-and-test approach that detected intersection of each inspection angle against each surface for every inspection point. Heuristic search was conducted through all possible combinations and a minimum set of accessible angles for the entire inspection was obtained. Nevertheless, the proposed method was trivial since it requires significant computational power to carry out the detection once the number of involved surfaces is considerable.

Many researchers pursued the above approach afterwards, which involved two approximations: (1) considering only point accessibility rather than entire feature accessibility; (2) concerning probe orientation domain as a set of discrete directions depending on the resolution of the mechanism of the probe head. With the objective of determining the approximate solution space, a generate-and-test strategy was widely adopt to find all feasible orientations by examining the accessibility of every discrete direction within the working environment.

Limaiem and Elmaraghy (1999) applied ray tracing as well on accessibility analysis for inspection points in the proposed inspection planning system. The probe was abstracted as several cylinders and therefore the actual probe size was taken into account while carrying out the ray modelling. Having examined the interference of the ray and the environment for all orientations of the probe, feasible directions were retained and stored in a binary matrix for the subsequent clustering process. Based on the results of the accessibility analysis, an optimization search technique was adopted to determine the sequence of measurement and resources allocation e.g. instrument set-ups, probes and probe orientations, according to criteria including minimum probe used, minimum changes of probe, and minimum changes of probe orientation.

While research interest being attracted on the accessibility analysis for discrete inspection points, Ziemian and Medeiros (1997; 1998) applied a projection methodology determining the accessibility of the entire feature and ultimately coping with orientating the workpiece on a CMM although the solution space was still discrete according to the resolution of the probe. Having assumed the absence of related fixturing and the use of indexable probe on only planar surfaces, their method examined the feasibility of a number of probe orientations, which were selected based on pre-defined strategy. Figure 2.13 reveals the feature accessibility analysis procedure proposed in the research. Initializing from the calculation of local feature accessibility with respect to abstracted probe vector, global point accessibility analysis was carried out at each of the feature vertices, which represented the feature boundary approximately. Any existence of intersections between the probe vector and facets of workpiece was detected by the calculation of line-plane and plane-plane intersection. Acknowledged intersection triggered the probe adjustment algorithm to search for a collision-free orientation. By contrast, a probe vector without intersection was then engaged with further iterated collision detection/adjustment considering all probe approximation e.g. vector, cylinder and rectangular block. The full-feature accessibility analysis was finally conducted resulting in every probe orientation from last step with its percentage of accessible surface area. A similar projection approach was used by Wu et al. (2004), which took the probe length and volume into account by means of projecting the probe stylus and body on the measured slots or holes.

Alvarez et al. (2008a; 2008b) and Fernandez et al. (2008) recently presented a methodology (Figure 2.14) for determining the probe orientation on CMM and laser direction on laser scanner based on discretized inspection part using stereolithography (STL) format. Visibility cone was sampled according to the device setup e.g. 673 orientations for the tested probe 720 orientations for the chosen laser system and intersection of each orientation with triangulated parts was detected by means of recursive ray traversal algorithm. Space partitioning technique was implemented to reduce the number of intersection tests by confining the boundary of participated facets on the part. Coupled with Back-face culling, the intersection check process was applied to the reminding facets with respect to each orientation and a binary value was assigned to this entry in the orientation matrix indicating the visible/non-visible status. Refined visibility analysis was carried out considering the real shape and dimension of the device modelled using simple geometrical shapes, which was underpinned by bounding volume theory that states the absence of collision between the objects within the bounded volume can be assured by the absence of any existing intersection between these bounding volumes. With the aim of minimizing the orientation changes, an algorithm was given to obtain a minimum number of clusters capable of inspecting every inspection point in the assignment.

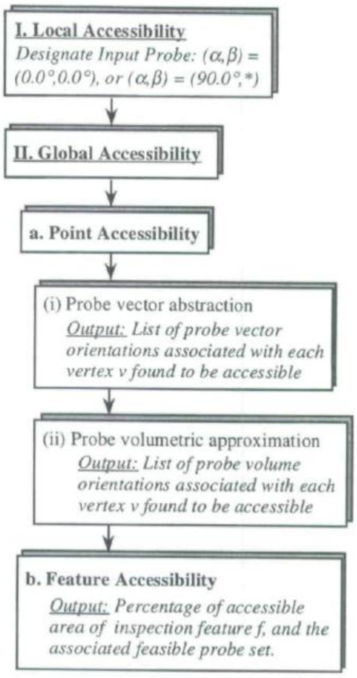


Figure 2.13 Methodology of the feature accessibility analysis (Ziemian and Medeiros, 1997).

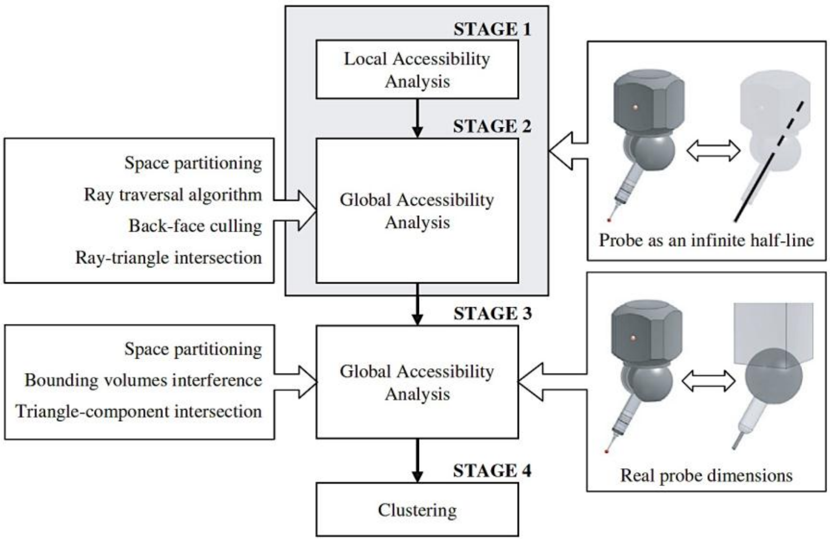


Figure 2.14 Accessibility analysis system overview presented in Alvarez et al. (2008a; 2008b)

2.6.2 Systems that compute exact solution

Apart from the approximate method analysing the accessibility, there were substantial research efforts focusing on the computation of exact solution. Visibility maps (VMs) was first introduced by Kim (1990; 1995) with comprehensive properties (Kim et al., 1995) and formally defined by Woo (1994) as a data structure to facilitate automation of toolpath generation in NC machining (Chen and Woo, 1992; Chen et al., 1993) as well as selecting part direction in mould and die design (Chen et al., 1993). By computing the dual image of the Gaussian map on the mapped unit sphere, locally complete visibility was obtained for specific surface. Local visibility of a point on a convex hull was defined by the hemispherical region on related Gaussian map with the normal vector as the pole. The local visibility map was then constructed by intersecting all the VMs of the points on the surface. Corresponding algorithms for separating and intersecting VMs on the unit sphere was accompanied with the solution determining the locally optimal cluster that intersected the maximal number of maps. Nevertheless, this method is only valid for applications not concerning global visibility on polyhedron surfaces. Figure 2.15 illustrates the Gauss maps for plane, cylinder and hemisphere with corresponding visibility maps respectively.

Limaiem and ElMaraghy (1997) proposed a generalized method for computing the continue accessibility domains based on the intersection of concentric spherical shells. Having applied geometric transformations, namely spherical scaling and solids intersection, intersection was implemented between the complement space of the part and a spherical shell that had the probe length as external radius and the maximum abstracted probe length as internal radius, which maximized the accessibility domain. Iteration of this intersection was continued with deceasing thickness of the shell until the desired precision was achieved. Common accessibility domain could be derived by comparing the domain of each point.

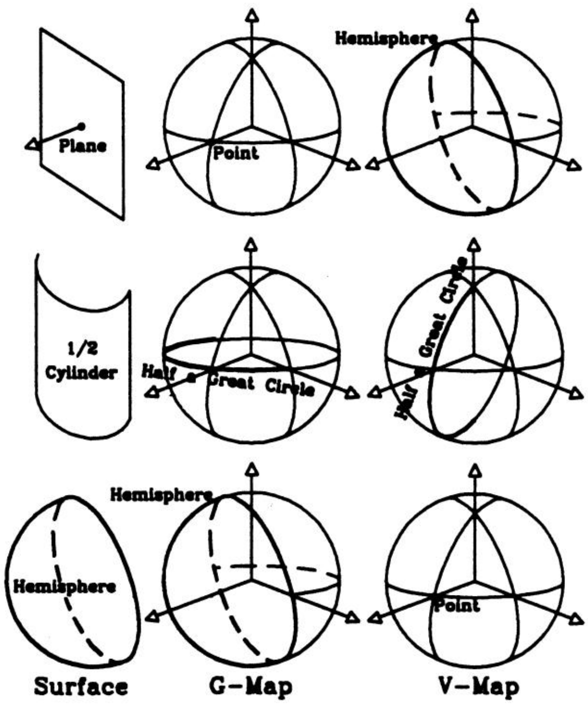


Figure 2.15 Examples of Gauss maps and visibility maps (Woo, 1994)

Similarly, though adopting an opposite manner, Jackman and Park (1998) proposed a method for finding the accessible orientations of the probe using constructed approximate VM. An arbitrary small hemisphere was initially located at the measured point and increased at a fixed step size that determined the accuracy level. A set of spherical caps was obtained by subtracting the workpiece model from the series of hemispheres generated in the iterations and the final VM was computed by normalizing all the points in caps to a unit sphere. The method can be simplified using visible vertex to derive the polyhedral visibility cone when the object is consisted of piecewise flat surfaces but the computational time is enormous if general object is considered.

Kweon and Medeiros (1998) introduced the concept of dimensioned VMs for determining the part orientation on CMMs with associated part description and tolerance information. Tolerance feature were analysed resulting in three types of surface taken account into the accessibility analysis: tolerance specified surfaces, datum related surfaces and interfering-surfaces with respect to the measured surfaces. The given algorithm searched for the potential interference surfaces starting from the adjacent surroundings and then the entire space based on proposition regarding accessibility in convex hull and different solids. VM clustering was accomplished by organizing the dimensionality of generated VMs e.g. point-VMs, line-VMs and polygon-VMs into different groups, which were assigned with the optimal orientation respectively.

Vafaeesefat and ElMaraghy (1998; 2000a; 2000b) presented an algorithm to analyse accessibility for CMMs and machining process by first determining the point accessibility domain (PAD) using the hemisphere on the unit sphere that contains the normal vector of the surface and then projecting all obstacles on the unit sphere. With a further projection of local accessibility and visible projected facets to the tangent plane at the point, the process of calculating the complement part on the unit sphere was transferred into 2D, which saved the vast computation needed in 3D. A binary Boolean operation based on image processing technique was employed to facilitate the generation of the final accessibility map and the feasibility of any probe direction can be checked by comparing the projection of the selected direction with the PAD. Although the proposed approach was able to achieve continuous point accessibility in 2D, a set of all possible discrete orientations were obtained due to this passive generate-and-test strategy. A clustering algorithm was supplied to find the optimal probe orientation according to user-defined constraints and corresponding rules were laid coping with the situation that multiple solutions emerged with the same minimum number of probe orientation.

Yang et al. (1999) carried out manufacturability analysis at the target point on sculptured surface by computing the visibility cone. Their methodology first determined the non-visible directions constrained by neighbouring surface patches based on the calculation of supporting cone with respect to the convex hull of those patches. Complementing non-visibility cone from local visibility cone was carried out afterwards for every surrounding surface until the final global visibility cone was generated.

Yin et al. (2000; 2002) presented a similar algorithm for computing visibility cones (VCs) by subtracting C-obstacles from the Configuration Space (C-Space). In order to handle the huge number of complementation of C-obstacles, visibility-culling techniques such as view-frustum culling, back-face culling and occlusion culling were utilized to eliminate non-visible facets to the viewer. Both partial visibility and complete visibility were able to be derived from the basic VCs of a point on a polyhedral object and applications on mould parting, NC-machining and CMMs probe orientating were given by the authors.

Dhaliwal et al. (2003) investigated the way finding the exact inaccessible region caused by the occlusion between a pair of triangular facets on polyhedral object. Rather than attempting to compute the infinite number of spherical projection of the occluded facet at every point on the target facet, the algorithm only computed the spherical projections on three unit spheres at vertices of the target facet resulting the convex hull of these three images, which was proven to be equivalent to the mathematical definition of the inaccessible region. Accessibility matrix was then constructed initially representing the accessibility of all facets from a set of directions and accordingly updated by examining the effects of other non-convex-hull facets in the same concave region. Another approximate algorithm was given to overcome the numerical difficulties using a fixed resolution while partitioning the unite sphere instead of modifying the resolution adaptively in the former. A variety of information regarding global accessibility were able to be obtained by querying the matrix including generating the GAC for a facet, identifying facets accessible from certain direction, searching facets occluding specific facet and comparing the amount of accessibility for different directions.

Liu and Ramani (2007) considered the construction of spherical visibility map (SVM) for convex facet using a similar manner that calculated the exact occlusion region due to another convex facet on a unit sphere. Different facing conditions of convex facets on a closed polyhedron were explored with defined properties. The occluded region was derived by finding the spherical convex hull of all spherical points describing the extremely stabbing lines and the SVM of a convex facet was obtained based on the subtraction of the unit sphere and occluded regions caused by all other facing facets.

With the same objective of defining non-visibility cones but from a different perspective, Li and Frank (2007) proposed a method to determine the boundaries of a non-visibility cone, denoted as sliding planes, for an arbitrary convex planar polygon occluded by other obstacle polygons. Emitting from the base facet, a 3D volumetric light beam was used to trace the boundary of side edges from both the obstacle and base facets thereby forming an occluded region on the unit sphere.

In order to tackle the intricate spherical arrangement calculation for convex decomposition, Liu et al. (2009) extended the above researches with the concept of Global Occlusion Map (GOM) in addition to the traditional Global Visibility Map (GVM), Local Visibility Map (LVM), which represented the set of total occluded view directions considering both self-occlusion and occlusion by other obstacles. Given detailed criteria, Obstacles were categorized into neighbouring occluder and potential occluder according to the geometric relation between selected surface and other surfaces. The GOM was obtained by calculating the spherical projection of Minkowski sum between the refection of the surface through the origin and all obstacles. As far as the Minkowski sum of two polyhedral surfaces was concerned, only a subset was calculated whereas it shared the identical spherical projection of a complete Minkowski sum, which resulted in a significant depletion of computational power. It was mathematically proved that the GVM of a surface could be derived by complementing the GOM from the LVM, which was a hemisphere with the pole orientating at the open space.

2.6.3 Systems utilizing graphic hardware

Balasubramaniam et al. (2003) described a technique to compute the visibility with reasonable efficiency for generating 5-axis NC roughing path based on the classic hidden surface removal algorithm. Not only was the measured surface tessellated to carry out the visibility check for individual triangle, but also the view space, Gaussian sphere in this case, was divided into eight patches and then subdivided into triangles hierarchically at designed resolution. The colour mapping process was then taken place to assign distinct red, blue and green (R,G,B) values to every triangles, which were subsequently converted into 24-bit numbers. This identification manner was capable of handling 1.6 million triangles if 8-bits colour graphics system was used. Eventually, the visibility of each triangle at specific orientation was obtained by querying the colours on the pixel map after rendering the scene using a depth-buffer.

Khardekar et al. (2006) presented algorithms taking advantage of modern graphics processing units (GPU) to detect any undercuts in identifying feasible mould parting directions. Using query function available on recent graphics cards, the system analysed the accessibility of the mould parting direction by checking if any pixels were rendered in current pass efficiently while reading frame-buffer instead of stencil-buffer. Another convex hull intersection algorithm was given to find all undercut-free parting directions on the Gaussian sphere. Inaccessibility region was defined and a new criterion was proposed to identify the potentially interacting pairs of facets, which eliminated 10%-90% of facet pairs before the convex hull intersection process. After the calculation of convex hull using Graham's scan algorithm, all vertices of line and their intersections obtained by projecting convex hull on a hemi-cube faces were tested for undercut-free check and feasible directions were generated as the output of the system.

Based on 3D digitization of the inspection part, Spitz et al. (1999) and Spitz and Requicha (2000) achieved fairly accurate results with a fast and efficient method when applied computer graphics hardware to generate GACs for both straight and bent tactile probes on CMMs. Cubic maps and depth-buffer were used to perspective project the obstacle as an approximation of the spherical projections as well as calculating the intersection and complement of a solid with a sphere. By growing the half-line that abstracted a straight probe, a semi-infinite cylinder with a hemisphere over the base was obtained and served as an approximation of the probe volume. Feature GAC was derived according to GACs of specific discrete points sampled on that feature and again cubic map instead of Minkowski sum was engaged to accomplish the intersection, which saved the expensive computation. An example is given in Figure 2.16 showing the generated GAC of a point on the CAD model using half-line probe abstraction and grown half-line probe respectively. Nonetheless, given those applied approximations, the proposed approach compromised accuracy and correctness for speed of the operation.

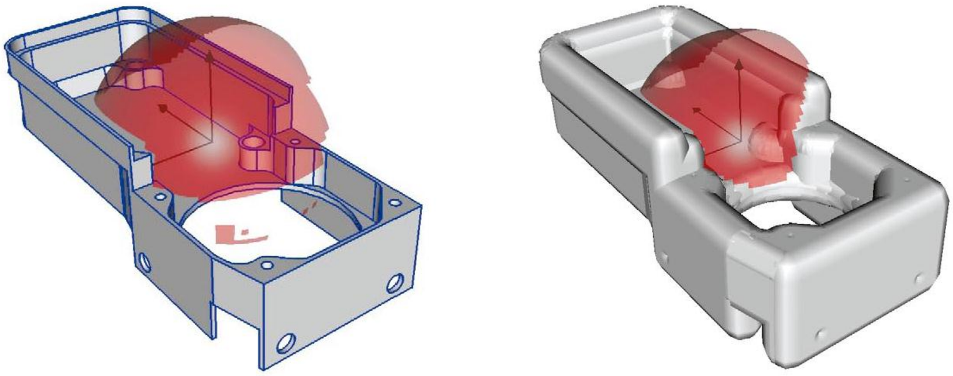


Figure 2.16 Obtained GAC of target point based on the CAD model (a) half-line probe (b) grown half-line probe (Spitz et al., 1999)

2.7 Clustering analysis

Throughout the literature, only research regarding clustering analysis for inspection carried out on CMMs can be found. In an inspection planning system for CMMs, it is clear that the objective of clustering the inspection operation is to group and sequence the target points and features. Subsequent planning process such as generating the probe path can then be carried out based on the clustering results. As the movement of CMMs is controlled by motors in different axes, the relative costing process is modifying the orientation of the probe and changing the probe rather than moving the probe once the part is mounted. The prior aim of clustering is therefore to minimize the changes related to the probe and as a result an optimized clustering algorithm leads to better inspection planning outcome in terms of efficiency and cost as well as potential impact on the accuracy. Given the significance of this process, although most researchers attempted to develop their clustering methodologies for CMMs in order to achieve the optimum group and sequence of inspection operations, this global optimization still remains a vital challenge due to the vast number of alternative resources and co-existing clustering scenarios.

Despite the emphasis on this particular step in most proposed inspection planners, the fundamental methodologies adopted by different researchers are surprisingly consistent. Depending on the chosen strategy of accessibility analysis, two major types of approaches are found throughout the literature and revealed in the following section respectively.

2.7.1 Clustering algorithm based on the discrete solution space of accessibility analysis

As mentioned in the previous section, the solution space of accessibility analysis for CMMs can be discretised based on the resolution of the mechanism of the selected probe head. The result is a set of probe orientations for every inspection point or feature, which are expressed by two rotating angles along the probe axes. The clustering process can be accordingly stated as:

Given a set of inspection points $\{\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_n}\}$ and a feasible orientation set $O_i = \{O_{i1}, O_{i2}, \ldots, O_{im}\}$ for each individual point $\mathbf{p_i}$, group these points into a minimal k clusters C_j with corresponding orientation set O_j, which is the superset of O_i . $\mathbf{p_i}$ should be included in at least one cluster while k is minimal.

Lim and Menq (1994) solved this problem by conducting an optimal angle search algorithm, which determined a set of probe angles according to the required inspection. The optimal angle was defined based on two contradictory aspects: (1) the angle should ensure a safe distance from the probe to the part, where collision might occur due to the misalignment of the part; (2) the amount of probe rotation should be kept at minimum level thus shortest probe rotation time and probe clearance time was induced. A heuristic method was adopted in their research to cope with the potential huge number of probe combination. In order to take advantage of the computational power from computer, the probe orientation and the inspection point information was codified as a pointer-list array where probe orientation constructed the pointer and inspection points constituted the list. The efficiency of a probe orientation was determined with respect to (1) the number of inspection points that can be inspected; (2) the grouping of the inspection points; (3) the number of inspection points in each group. A predefined sorting routine was then activated to sort the list resulting in a column of orientations ranking with the corresponding efficiency. A minimum angle set was created based on those most efficient orientations and each orientation was filtered with the defined criteria sequentially to obtain the minimal number of angles. One drawback of this method was the possibility of multiple combinations of probe orientations for the same input and no further solution was given by the authors.

Aiming to obtain higher inspection accuracy with low inspection time and cost, Ajmal and Zhang (1998) proposed a knowledge-based clustering algorithm for inspection planning of CMMs. A variety of group technology (GT) was reviewed to select a suitable method such as matrix formulation, mathematical programming formulation, graph formulation, and knowledge-based and neural network-cased formulation. Knowledge-based cluster algorithm was eventually adopted since the computational time complexity $O(mn)$ is lower than the traditional methods $O(m^2n+n^2m)$. One common requirement of implementing GT methods is the construction of incidence matrix. Therefore the accessibility cone of a probe was sampled into a set of discrete probe orientations and the relationship between inspection probes and features was indicated by the developed incidence matrix, where the probe orientations were represented by the row and the tolerance features were represented by the column in order. The element of the matrix was assigned value 1 if the specific tolerance feature can be inspected using the certain orientation and 0 if the orientation is not applicable. With the objective of grouping tolerance features using the same probe orientations, both rows and columns were rearranged to construct the block diagonal matrix. Clusters were accordingly generated and the exchange of probe was minimized within the cluster, which gives the same installation datum and minimal calibration error. Table 2.3 (a) and (b) show the example of this approach. After the rearrangement of the matrix, features 2,3,5,8 were grouped together and the orientation *Z-* was selected to perform the inspection of these four features. The rest features were clustered using the same method and one may notice that there were two available orientations for feature 1 and 6 where no further explanation was given. The application of the proposed methodology was however severely limited since only six directions along three axes were considered while the accessibility cone was being sampled.

Zhang et al. (2000) improved this algorithm by incorporating the weight factor assigned to different orientations to avoid potential collision. Weight factor 1.0 was assigned to the selected probe in its optimal orientation and weight factor 0.25 was assigned otherwise. However the process of identifying the optimum direction of the probe was not given in detail.

Table 2.3 Knowledge-based clustering algorithm for inspection planning of CMMs (Ajmal and Zhang, 1998)

(a)

	Tolerance feature identities							
Probe orientations	1	2	3	4	5	6	7	8
X+		1	1		1			
X–	1					1		
Y+				1			1	
Y–	1					1		
Z+				1				
Z–		1	1		1			1

(b)

	Tolerance feature identities							
Probe orientations	2	3	5	8	1	6	4	7
X+	1	1	1					
Z–	1	1	1	1				
X–					1	1		
Y–					1	1		
Y+							1	1
Z+							1	

Limaiem and ElMaraghy (1997; 1999) proposed an optimization algorithm to cluster and sequence the probe orientations with respect to the inspection part. A new term, principle cluster, was defined as a probe with a set of orientations to represent the Tool/Operation matrix, which enabled the implementation of the classical sequencing and resources planning. The clustering process was carried out based on the results of the accessibility analysis and two scenarios were taken into account: (1) clustering with precedence constraints e.g. the priority of important features in terms of function and accuracy, the priority of tolerance types, commonality of datum and the configuration of reference frame and part localization; and (2) clustering without precedence constraints. For clustering with precedence constraints, optimization was carried out to minimize the resource change and keep only one resource to every operation by fixing the first assignment and then select subsequent assignments that shared the same resources whereas no precedence constraints were bounded. This process was applied for each possible first assignment and evaluations of all alternatives were generated to choose the global optimal sequence. For clustering without precedence constraints, a search algorithm was engaged to determine the largest set of assignments that shared the same resource and removed from the active assignment list, which initially included all assignments. The process was iterated until all assignments were at least incorporated in one list and repeated in sequence for different resources if multiple resources were considered.

Vafaeesefat and Elmaraghy (2000a) proposed a binary matrix to represent the discrete accessible probe directions with respect to the measurement points using value 1for feasible orientations and 0 for non-feasible orientations. By projecting all probe orientations onto the 2D accessibility region of the target measurement point, obtained nodes with value 1 created the accessible domain of the point. It can be concluded that measurement points can be inspected using the same orientations if and only if their accessible domains intersected. Clustering process was then conducted by means of calculating the maximum intersections among different accessible domains. An algorithm was proposed searching through all points for the largest set of common orientations and a number of clusters were finally generated without any intersection of accessible domains. In order to obtain a unique solution of the clustering, three constraints were injected into the search process: (1) the clusters covered all measurement points except the points with no accessible orientation; (2) the number of clusters was kept to be minimal; (3) each cluster contained the maximum number of accessible orientations. Moreover, the solutions were assessed to find the orientations at the safest distance from potential obstacles to avoid collision caused by parts misalignment.

Fernandez et al. (2008) applied a similar algorithm for laser scanner inspection planning to cluster the triangles that discretized the part based on the maximum common scanning directions. A binary matrix was created to indicate the accessibility of each triangle against different scanning orientations and a search process led to the results that comprised a set of clusters with maximum intersection of the visibility cones. Alvarez et al. (2008a) extended this method to accessibility analysis for CMMs inspection planning successfully and the effectiveness was demonstrated by several case studies.

2.7.2 Clustering algorithm based on the exact solution space of accessibility analysis

There were only a few researchers who attempted to determine the clusters based on the exact solution space. Spyridi and Requicha (1990) considered this a minimal set problem for all GACs. The minimal set of accessible probes could be computed by classifying all GACs into a minimal set clusters and then intersecting GACs in the same cluster to define the common direction zones. Any combination of directions from each zone could subsequently fabricate the inspection plan. Since the minimal clustering problem was NP-complete, no methodology was given in the paper but discretizing the GACs using a grid of spatial cells was suggested to find the common spatial region by comparing the cones that intersected the same cell. The uniqueness of the solution was well addressed and factors such as workpiece stability, setup cost, and fixturing were considered critical to generate the most efficient solution although only simple heuristic evaluation function was employed in their study.

Woo (1994) presented an effective way to orient the workpiece after determining the VMs for each surface, which was equivalent to finding the minimum number of points or relevant spherical polygons commonly comprised by the corresponding maps. Finding the global optimum solution of this problem was deemed as a set-covering problem and also NP-hard. A locally optimal solution algorithm was however put forward using methodologies for separating and intersecting spherical polygons proposed by Chen et al. (1993). With the aim of finding a hemisphere containing the largest number of polygons in a given set, the algorithm was able to determine the densest hemisphere by computing and clustering the ownership vector of each spherically convex polygon. The problem was then converted into finding the great circle on the unit sphere that intersected with the greatest number of maps, known as sandwich cutting. Figure 2.17 shows the incircle approximations of a VM for 12 surfaces of an object with corresponding labels and Figure 2.18 gives the analysed result with the great circle intersecting all maps except M_{12} shown in broken line. Clusters were created based on the intersection among all VMs listed in Table 2.4. It can be seen that only two setups are needed to machine all the surfaces since M_1-M_{11} are penetrated by one great circle. Nevertheless, there was no approach for selecting a specific orientation of the part since multiple applicable great circles could be found.

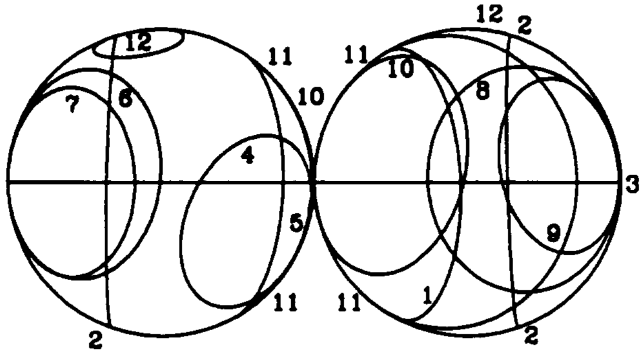


Figure 2.17 A VM for 12 surfaces of an object

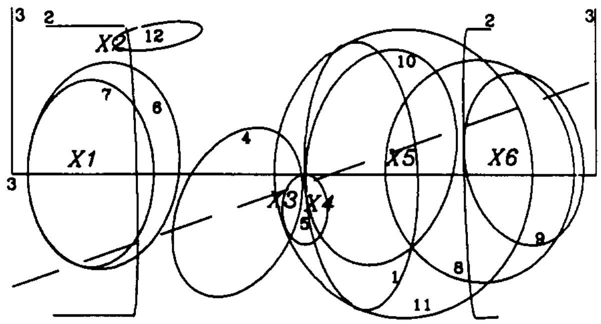


Figure 2.18 Sandwich cutting process

Table 2.4 Clustering results of the VM

Map	Cluster					
	x_1	x_2	x_3	x_4	x_5	x_6
M_1				1	1	
M_2	1	1				1
M_3	1				1	1
M_4			1			
M_5			1	1		
M_6	1					
M_7	1					
M_8					1	1
M_9						1
M_{10}				1	1	
M_{11}			1	1		1
M_{12}		1				

Kweon and Medeiros (1998) defined a clustering heuristic to process and group the generated VMs including single point-VMs, double points-VMs, line-VMs and polygon-VMs. This graph partitioning based method used augmenting and disjointing algorithms to eliminate redundant VMs as well as consolidating higher dimensioned VMs to lower dimensioned VMs, which simplified the final cluster computation. The process initiated from screening all point-VMs while identical ones were removed and double point-VMs were replaced by single-VMs included by the former. Line-VMs and polygon-VMs that comprised all remaining point-VMs were then positively augmented since their associated tolerances could also be accessed by direction those point-VMs presented. The rest line-VMs and polygon-VMs were positively augmented using the same manner. For polygon-VMs that comprised no line-VMs were negatively augmented with corresponding intersecting line-VMs, which resulted in a smaller spherical line segment. Intersecting VMs disjointing algorithm was conducted afterwards according to both the number and the magnitude of the intersection. In order to converge the clustering efficiently, VM that had only one intersection was grouped together regardless of how many intersections the other VM in the pair had. Maximum magnitude of intersection was desirable while the remaining VMs were clustered. The decision of selecting the best orientation was left for the inspection planner to achieve reasonable flexibility of the planning system.

2.8 Measurement path planning for CMMs

It can be found throughout the literature that substantial research efforts have been invested in the path planning activity generating the trajectory for the movement of the CMM probe (Zhao et al., 2009). It is crucial to generate a path that not only maximizes the efficiency but also avoids collision between the probe and the parts. Two characteristics of the path planning for CMM can be concluded:

1) Since the accessibility analysis has been conducted, the path planning is for points that can be measured in a certain orientation with a specific probe. The sequence of different orientations was not taken into account for path planning.

2) By considering the CMMs as Cartesian robots, collision detection and avoidance attracted the primary attention of most path planning systems, which resulted in a collision-free path but not necessarily optimal.

Lim and Menq (1994) proposed an approach to avoid collision between two inspection points by adding several safe-points into the approach path, the retract path and the drive path, which were the three main paths consisted of the probe trajectory. The probe orientation information was first included to group points that shared the same angle. This angle was also considered the safe-angle for both the probe stylus and the probe head to access the points. Safe-points were added along this angle and as a result the pre-approach path and pre-retract path were generated to ensure that no collision could occur due to the rotation of the probe head. The approach heavily relied on manual operation and the generated path still must be evaluated by path verification (Yau and Menq, 1991; 1992)

Space subdivision methods such as spatial enumeration and octree database were widely applied for robot path planning. A 3D octree ray tracing approach was proposed by Lu et al. (1994) to obtain an optimal path between two consecutive inspection points as a complementary process of measurement sequence generation. Objects that were polyhedral solids were represented in octree data structure while non-polyhedral objects were replaced by the minimum enclosing polyhedral surfaces before applying the proposed algorithm. In order to cope with the zigzag nature of the path inherent with traditional octree methods, global information regarding the obstacle vertices was incorporated, which benefited the dynamic performance of the planning system by reducing the acceleration and deceleration of the probe. The detection of obstacle was carried out based on a ray tracing technique only examining the obstacles penetrated by the tracing ray. The authors claimed that faster and simpler detection and optimization result were achieved by observing a significant improvement of inspection time.

Despite the considerable complexity of the computation, Limaiem and ElMaraghy (1999) adopted Dijkistra's shortest path in their CATIP system aiming to determine the shortest path between two measuring points. Both static and dynamic interference were considered in their study and approximate representation of the probe, the part and the surroundings were employed to avoid expensive computation. The workspace was discretized using uniform Cartesian mapping based a heuristic formula decided by the dimension of the probe and the part and surroundings were simplified using bounding volume, in this case octree decomposition. The approach path, the retract path and the drive path as well as the safe-points were used to check the interference after the approximation. By evaluating the interference between bounding volumes of the probe and the part, a collision-free path was subsequently generated, which was optimal for this specific discretization.

A general framework of generating the probe path for CMM using any CAD system was proposed (Lin and Murugappan, 1998; Lin and Mahabaleshwarkar, 1999). With the aim of providing a generic algorithm that can be applied with any CAD system API, the probe was approximated by a single point object converting the detection of the probe with the object into the detection of collision of a single point with the object The surroundings including fixtures of the object were also ignored to maintain the simplicity and the starting point of the probe must be specified by the user. Having linked the system with CAD system using specific API, the target point and involved entities were selected by the user. The ray tracing technique was employed initiating from the starting point to the target to detect the collision between the imaginary ray and the object. If interference was found, every face on the entity was searched starting from the adjacent faces until the target point was located. A path on the entity was then generated by connecting the midpoints of the common edges of all involving faces. However, the application of this method was severely limited to simple objects only containing planar surfaces and the obtained path was not optimized.

Albuquerque et al. (2000) developed a system capable of automatically placing inspection points and generating collision-free inspection path for complex objects with multiple intersecting features. By implementing an iterative method, a sufficient number of inspection points were placed on each inspection surface to acquire the geometric information of specific features. The accessibility of each point was then checked by bounding the probe with its accessories into a sphere. In addition, an extra position of the point that is a user-defined distance away from the surface was examined for accessibility as the rest position of the probe. Moments and a search routine were applied in order to represent the local region by a set of points evenly. For points on convex curved surface, via points were added manually to avoid potential collision. The path on the same surface was finally generated based on a nearest neighbour search algorithm, which was adopted to sequence the inspection surfaces as well. However, the located inspection points might not represent the surface accurately if only a certain region of the part was sampled. The approach could only plan for parts with simple shapes whereas more complex shapes such as free-form surfaces were not considered.

Ainsworth et al. (2000) presented a methodology for inspection planning and collision-free path generation on complex engineering component geometry, namely free-form shapes. After registering the actual part with the corresponding CAD model and sampling the surfaces using a grid of discrete points, the developed programme generated a inspection path for the selected entity based on user-defined parameters including scan direction, scanning pattern, stand-off distance, probe head orientation. It was claimed that unidirectional scans suited for closed and/or highly folded surfaces while bi-directional scans resulted in better representation of relatively flat and open surfaces. The user also was in charge of manipulating the path interactively by observing the interference between the path segments and the 3D model of the part. A probe path verification process was the final stage of the system to evaluate the generated path visually by means of simulating the inspection using full 3D graphical animation technique. A complementary automatic collision detection algorithm was incorporated in case the visual verification was incapable of making the decision. The trajectory of the sphere centre was examined for collision with the offset model surfaces. Nonetheless, no details regarding the algorithm were given in the paper.

It is a common conclusion that in order to create a collision-free path a secondary position must be chosen to avoid the interference between the probe and the part, known as the dummy point, or guide point. Most complete inspection planner utilized this approach to implement the path planning with the assistance of the user interaction (Cho and Kim, 1995; Spitz and Requicha, 2000; Zhang et al., 2000; Cho and Seo, 2002; Cho et al., 2004; Cho et al., 2005; Mohib et al., 2009). Spitz and Requicha (2000) proposed a practical path planning system to tackle the sequencing and collision avoidance automatically. Probabilistic roadmap planner was adopted with the extension to multiple-goals, namely the inspection points. Every point was considered a goal and the objective was to find the shortest tour of the goal configuration, which was a closed path that visited each goal at least once without any collision with the part. The roadmap was initially constructed using goals as the nodes and paths as the edges. Random nodes were injected into the roadmap by the local planner in order to find the collision-free path between neighbouring goals. After the goal configuration was connected entirely, the optimal tour was extracted from all possible paths among the edges by applying the solution of the travelling salesperson problem (TSP). A number of CMM heuristics were given to assist the local planner to enhance the construction of the roadmap. Since the inserted nodes were random and might not be at the optimal position, the outputted tour was only optimal with respect to this specific goal configuration.

Despite the substantial research efforts in path planning for CMMs, the practical application was severely limited due to three courses:

1) All researchers constrained their approaches for a certain type of geometry. There is no method that can be applied on generic objects.
2) Manual interaction was heavily needed in most methods. Processes such as inserting the dummy points and sequencing different surfaces were conducted by user in order to obtain an acceptable efficiency.
3) The number of involved entities was limited as the check of collision detection consumed a significant amount of computational resource and time.

2.9 Research gaps

According to the literature, a number of research gaps have been identified:

1) The majority of the work has been done on inspection planning for CMMs. Early development of the approaches was limited by the computer techniques and therefore no realistic application could be accompanied by the developed systems. Most recent research tried to solve the problem using the same strategy including inspection tasks identification, inspection point generation and distribution, accessibility analysis, clustering analysis and path planning. However, no research was found for IPP system for LVM systems. A systemic framework is required to cope with the general inspection planning process for systems including laser tracker, laser radar, iGPS and photogrammetry.

2) GD&T was the primary concern when design requirements were interpreted. However, the process relies heavily on human interaction. A more effective way of collecting and representing the GD&T in metrology environment is needed.

3) Probe selection was only concerned in some research and not addressed as an important issue. Nevertheless, instrument selection is deemed as a compulsory and critical process in inspection planning for large volume metrology applications due to the increasing number of systems with different capabilities. This creates difficulties in selecting the most appropriate instrument for a given measurement task. A process is required that identifies capable and suitable metrology devices with respect to desired measurement tasks.

4) Accessibility has gained huge research interest for CMM inspection planning since it not only confined the probe orientation, but also laid the foundation for later planning stages such as sequencing and clustering. Unlike the CMM, LVM systems apply optical-based techniques and the line-of-sight replaces the orientation of the probe on CMM as the major concern of visibility. Therefore an approach focusing on the visibility analysis for LYM instruments such as laser tracker is mandatory. Moreover, corresponding algorithms are need for subsequent clustering process to be carried out based on the results of visibility analysis.

5) Surprisingly, the accuracy of the inspection was not emphasized in most studies. This is not acceptable for LVM inspection since the performance of the system largely depends on the relative location. A methodology that can determine the most suitable position of the instrument is required for achieving minimum uncertainty.

6) The applicability of those systems is severely limited due to the distinctive and highly self-contained data structures, which resulting in outputs not compatible with the interfaces communicating with specific instruments. Under this circumstance, an effective and efficient data exchanging between the proposed system and metrology software within CAD environment is desirable.

CHAPTER 3 RESEARCH OBJECTIVES, STAGES, DELIVERABLES AND OUTLINE OF THE NEW IPP FRAMEWORK FOR LVM SYSTEMS

3.1 Introduction

This chapter presents the specific objectives of the study together with the expected outcomes which are identified based on the requirements of a typical IPP system. The stages of the research are stated to describe the process of the study and the research outcomes with associated chapters are presented.

Based on the proposed objectives, a new inspection planning process framework for LVM systems is developed. The framework is intended to generate an optimised inspection plan based on the input of a prismatic part from a CAD model. The structure and functionalities of the system are described in this chapter with the assistance of Unified Modelling Language (UML) to clearly indicate the designed workflow and capabilities.

3.2 Research objectives

Based on the reviewed literature in Chapter 2, the objectives of this research are identified and presented in this section in order to accomplish the proposed research aim in Chapter 1.

A workflow for a product verification process is first defined using UML class diagram shown in Figure 3.1, which reveals the participating object classes with corresponding interactions. Initially, the product is created by the designer in the form of CAD model by means of 3D modelling software. The IPP system is expected to extract the GD&T information incorporated in the design and generate an inspection plan that can be interpreted by the metrology software. The metrologist thus conducts the inspection according to the guidance of the software that handles the data collection and conformance report generation concurrently. The function names are shown above the leads while the arrows indicate the assigned relationships between two objects. The permitted data accessibility is represented by the number shown next to the object. ‘0’ indicates no permitted access and ‘1’ indicates single access while ‘*’ means unlimited access. The reader is recommended to refer to relevant standards for detailed explanation of UML class diagram (BSI, 2008; ISO, 2012).

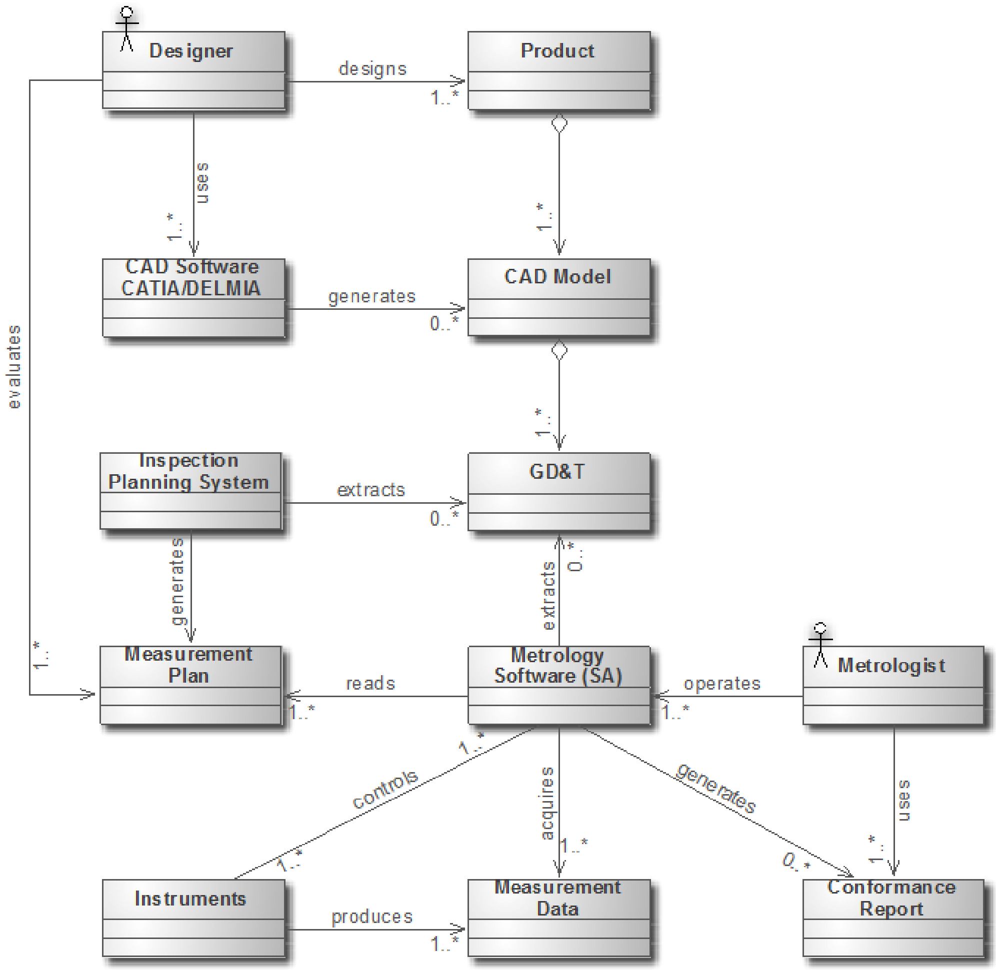


Figure 3.1 The modelled interactions among inspection activities

The author has identified a common drawback that exists in current IPP systems. The applicability of those systems is severely limited due to the distinctive and high self-contained data structures, which results in outputs that are not compatible with the interfaces communicating with specific instruments. Under this circumstance, an effective and efficient data exchange between the proposed system and metrology software within CAD environment is desirable. As shown in Figure 3.2, benefitted from this seamless integration, the user is able to carry out vital activities such as visibility check, optimized instrument locating, inspection simulation, and inspection execution, as extended functions to the metrology software.

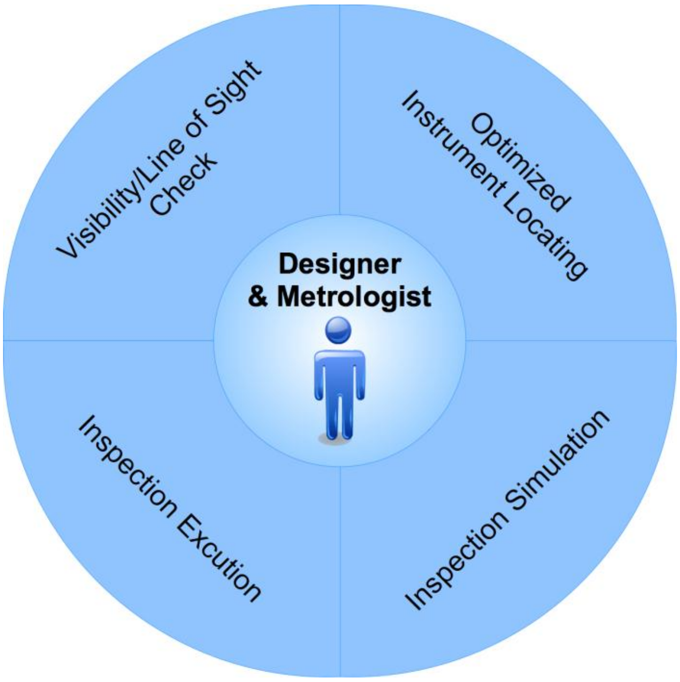


Figure 3.2 Individual functions supported by the proposed system

In order to implement the workflow in the proposed IPP system, the following objectives are identified:

1) Review the literature and state of the art in methods of inspection planning, and related algorithms within the scope of the thesis.
2) Develop a systematic theoretical framework for inspection planning for LVM systems within a digital modelling and planning environment.
3) Develop methods by which to comprehend the design in terms of inspection requirements and specify the measurement tasks and requirements.
4) Define new methodologies to determine the appropriate instruments based on predefined criteria and identify the measuring strategy.

5) Develop algorithms to configure the selected instruments according to visibility analysis and performance with regards to accuracy and for the optimization of the inspection operations to achieve minimal cost and time.
6) Verify the effectiveness of the framework by applying the methods and algorithms to both an artefact and an industrial component. Document the methods and disseminate the research outcomes.

3.3 Research stages and deliverables

To achieve the above objectives, the research stages are summarized as follows:

a) Review of available IPP systems and LVM systems

A thorough survey is carried out in the literature of IPP systems. The development of relevant methodologies and techniques is reviewed together with the latest applications on current inspection instruments, namely CMMs and laser scanners. Recent developments on LVM systems are also surveyed to identify the distinctive requirements for IPP systems. A brief review of present metrology software is presented subsequently to obtain a vision regarding the supported capabilities and potential integration options.

b) Develop a frame work for LVM IPP system

The desired functionalities of the system are specified based on the review. The structure of the system is proposed coupled with the workflow. Vital parts including the interface, core modules and supporting knowledge base are presented providing a complete process from extracting the design data to generating optimized inspection plan in specific codes.

c) Accomplish the designed functions in the system

Seven functional core modules are developed including tolerance feature analysis, instrument selection, inspection point selection, accessibility and visibility analysis, clustering analysis, instrument setup and configuration and measurement sequencing. Due to the considerable research diversity embodied in these processes, each module is accompanied with individual introduction and literature review. Examples and demonstrations are given to verify the effectiveness of each process.

d) Evaluate the proposed methodology through case studies

The developed system is validated using case studies involving generic parts. The processes employed in inspection planning are demonstrated respectively. The compatibility is proved by involving different CAD models and a variety of instruments.

The deliverables are given in Table 3.1 Research outcomes and deliverables with associated chapters with the associated chapters.

Table 3.1 Research outcomes and deliverables with associated chapters

Objective	**Chapter**	**Research Outcomes/Deliverables**
1	2,4,5,6,7,8	Literature and state of the art review
2	3	Theoretical framework for inspection planning process for LVM systems within the scope
3	4	Method of generating measurement task based on design specification e.g. GD&T
4	5	Algorithms for automated instrument selection
5	6,7,8	Algorithms for visibility analysis, clustering analysis and path planning for configuring LVM systems
6	9	Performance testing of the developed framework and algorithms

3.4 Proposed framework for IPP for LVM systems

Based on the aims and objectives stated above, the functionalities and the interaction between the functional modules and operations are proposed using UML use case diagram, shown in Figure 3.3.

The primary users of the system are the designer and metrologist (inspector). Traditionally, there is no available information regarding the quality and the validity of the assigned GD&T after the design process. By using the proposed system, the designer is given the ability of verifying the design with respect to the measurability of the product. In order to carry out the check, the system must select an appropriate measurement system for the tasks according to the analysis results of the GD&T. The measurability is then checked in four aspects including physical capability, accuracy capability, measurement cost and technology readiness level. This process also enables the designer to improve the design in terms of measurability e.g. relaxing tolerance on non-key characteristic features. On the other hand, the metrologist and inspector can use the proposed system to plan the measurement and inspection process. Based on the selected measurement system, the measurement features are first recognised and extracted from the solid model by analysing the GD&T. The measurement points are then selected and distributed on the measurement features based on parameters such as required accuracy and measurement speed. The clustering of measurement points takes place afterwards followed by instrument configuration, setup and calibration. The output of the inspection process planning function is a measurement plan with identified measuring path. Inspection simulation can be conducted based on the measurement plan to verify and validate the generated plan. Finally, the selected instrument is controlled by the proposed system to execute the measurement.

With the intention of providing the functionalities defined in Figure 3.3, a large volume metrology inspection system (LVMIS) is proposed in this study and the structure is shown in Figure 3.4. The system provides a complete process from extracting the design data to generating optimized inspection plan in specific codes.

The framework can be divided into three main parts:

LVMIS Interface: The workflow of an inspection planning process starts from the metrology software acquiring the design information from computer-aided process software e.g. CAD from CATIA and Digital Environment (DE) from DELMIA. Geometry of the product as well as the GD&T information is then available in the database of the used metrology software. The LVMIS interface manages the communication and data transfer between LVMIS and metrology software. On the one hand, it enables the proposed system to query the database of the metrology software and hence the geometry of the product with GD&T can be used for the core modules. In addition to that, existing functions in metrology software can be utilised directly by the system such as inspection simulation and inspection execution.

On the other hand, the system provides unique capabilities to any external program or software through the interface, including visibility check and optimized instrument localization. In the case that only certain function is needed, it is not required to complete the entire workflow in the LVMIS. The metrology software can carry out the relevant activities and output the results. The benefit of this will be discussed later in relevant chapters.

Core Modules: This core section consists of seven functional modules, which underpin the inspection planning process. Each module aims at different tasks and provides unique functionalities, which are defined as follows:

(a) Tolerance feature analysis extracts design information, namely GD&T in this work, from the metrology software by the tolerance feature analysis module through the LVMIS interface. As GD&T has become the most applied method for describing the nominal geometry of the part as well as its permitted deviation, it contains information regarding the importance of different features of the product and also the intention of the designer with respect to the functionality in assembly. Inspection tasks can hence be identified according to the decomposed GD&T information using specific data structure. Measurement features are outputted from this module coupled with allowed deviation.

(b) Inspection instrument selection module assesses the task requirements according to the output of the previous module and then chooses the most suitable instrument or instruments' bundle based on user-defined criteria such as volumetric coverage, accuracy, cost and technology readiness level. Since this is the beginning of the inspection planning where no exact inspection information is available e.g. measurement system, measurement strategy and measurement speed, the module should be capable of analysing and processing vague information and selecting appropriate instrument based on non-quantitative information. Inspection point selection module then determines the number and distribution of the sampling points on the target part based on international standards for the select instrument.

(c) Visibility analysis module carries out the generation of the spatial solution domain for each measurement point. It is essential to carry out this process in order to improve the effectiveness and efficiency of the inspection plan. Not only should the inspection part considered in the process but also the surrounding such as the instrument, gigs and fixtures must be taken into account in the analysis. Additionally, continuous spatial domain is required rather than discrete visible location as it compromises the full 3D measurement capability of latest LVM system. Visibility zones for individual points are subsequently outputted to the next module for clustering analysis.

(d) Clustering analysis modules calculates the intersection of all individual visibility zones from the visibility analysis module and then groups them into clusters that cover all measurement points. The number of the clusters must be minimised to eliminate unnecessary location change of the instrument as it leads to (i) additional time caused by re-calibration after relocation (ii) additional uncertainty caused by correlating multiple coordinate systems (iii) additional uncertainty induced by the alteration of temperature gradient.

(e) Instrument configuration module determines the best position in each clustered visibility zone. As visibility has been guaranteed in the solution space, the objective of this optimization is to minimise the total uncertainty to all measurement points.

(f) Ultimately, the measurement sequencing module derives the sequence of the measurement points in order to maximize the efficiency of the plan. As the measuring volume of LVM is considerably large, the inspection time and efficiency is most affected by the traversing distance of the measurement target device. Therefore, the optimal sequence can be obtained when the minimum accumulated length of the paths among all measurement points is achieved.

The output of the system is optimized inspection plan that can be converted to measurement plan (MP) codes and imported by metrology software to simulate and execute the inspection.

Supporting Knowledge Base: This support section comprises four knowledge base including: Inspection Features, GD&T Knowledge, LVM Instrument Database and Metrology Standards & Best Practices. Standards including decision rules (BS EN ISO14253-1, 1999; ASME B89.7.3.1, 2001), Geometrical Product Specifications (GPS) (ISO/TR 14638, 1995; BS EN ISO 1101, 2005), GD&T definition (ASME Y14.5, 2009) and sampling strategy (BS 7172, 1989) serve the core modules with specific guidance and knowledge rules, as well as assisting the decision making and standardizing the process.

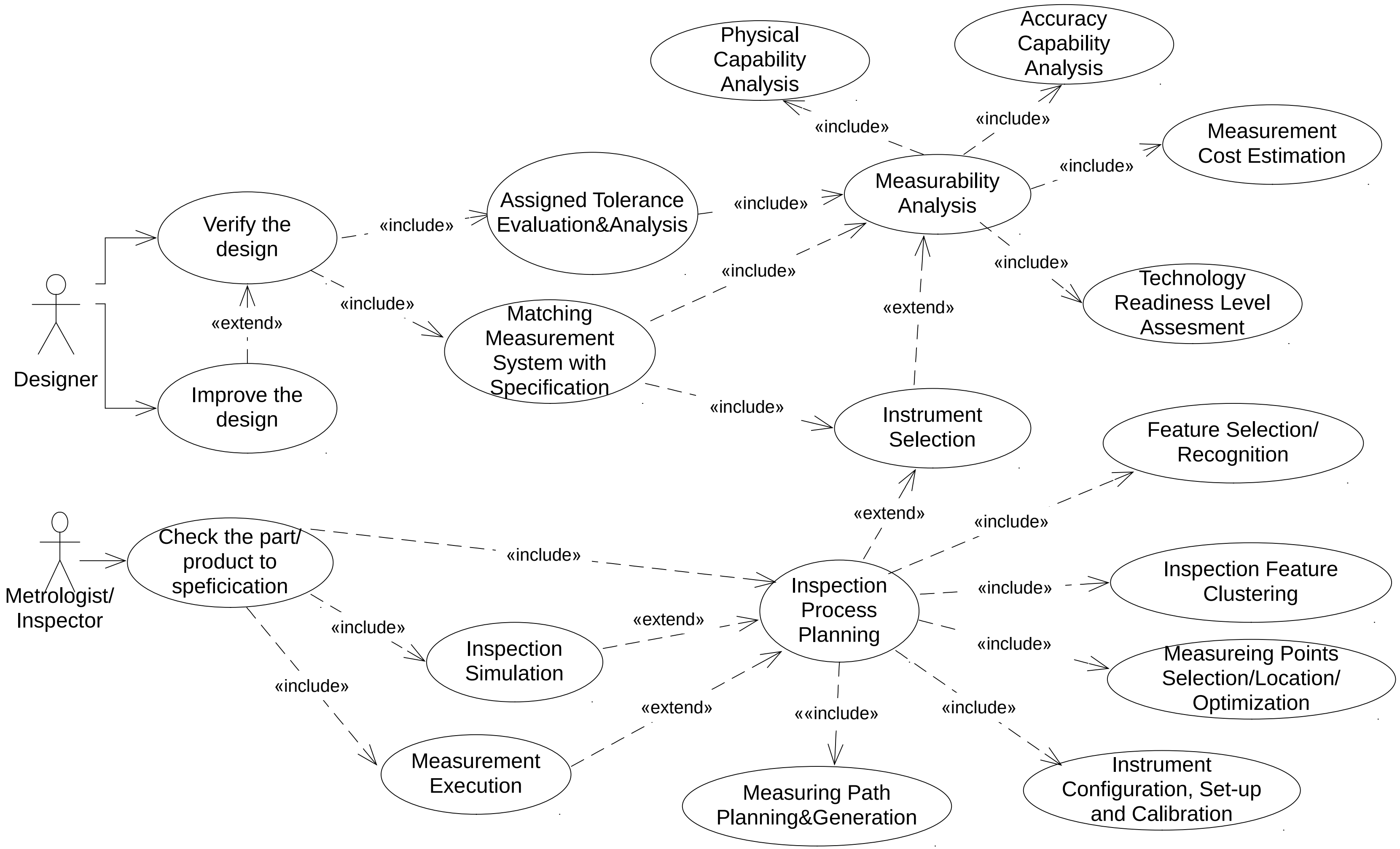


Figure 3.3 The use case representation of the framework

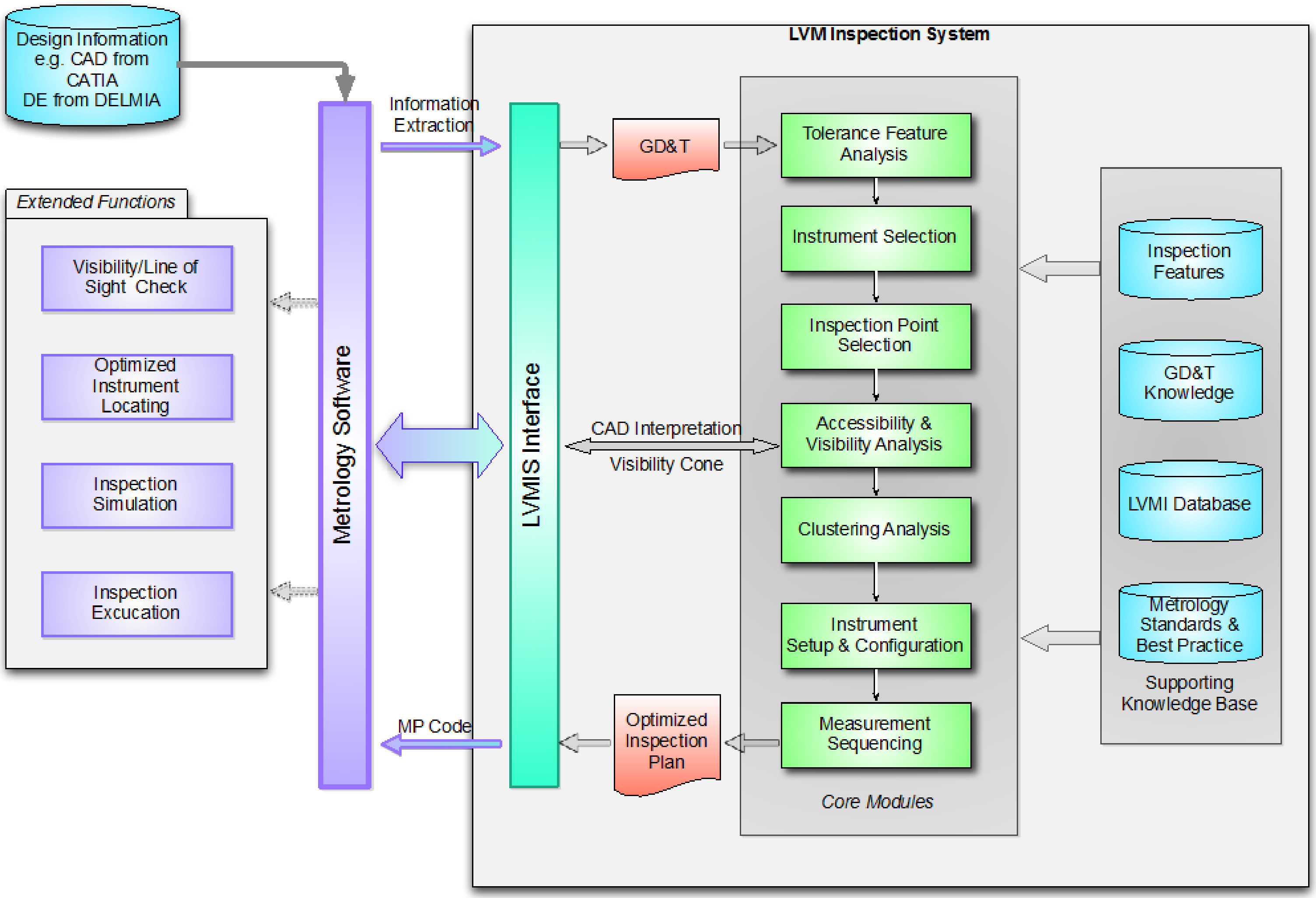


Figure 3.4 The proposed IPP for LVM system structure

CHAPTER 4 TOLERANCE FEATURE ANALYSIS

4.1 Introduction

In order to perform functionally, the product must satisfy the design requirements after the manufacturing process. As far as the geometrical dimension is concerned, the product should be within a certain range of deviation from the design model. However, with the increasing complexity of modern products, it is neither practical nor necessary to examine every inch of the part since the importance of the dimensional accuracy for the acceptance of the part varies significantly. Therefore, it is crucial for the operator to question what should be inspected on the product at the beginning of the planning process.

Geometric dimensioning and tolerancing (GD&T) was first standardized for the purpose of defining and exchanging information on engineering drawings by means of implementing a symbolic language (Wikipedia). Along with the rapid development of computer-aided design, the system soon became the most applied method for describing the nominal geometry of the part as well as its permitted deviation. Under this circumstance, the early research of inspection planning was predominantly driven by the GD&T, according to the review conducted by Li and Gu (2004) and Zhao et al. (2009). Nevertheless, inspection features and tolerance characteristic were considered and classified separately (Beg and Shunmugam, 2002; Cho et al., 2005; Mohib et al., 2009; Zhao et al., 2009) and feature recognition has attracted huge research interest and has dominated the way of specifying inspection tasks due to developed CAD techniques (Joshi and Chang, 1988a; 1988b; Helmy, 1991; Gao et al., 2004; Wong et al., 2006).

Despite some successful applications, they all face the inherent complexity and variety of features existing in modern manufacturing companies. The extraction system is considerably complicated with detailed knowledge or information base and only certain types of features can be extracted and decomposed. Modification and enhancement of the module is needed whilst any new feature emerges during the design stages. Characterized by the relatively large scale, inspection that falls in the large volume metrology range normally faces products with distinct features. Measuring the entire feature may lead to redundant data and additional inspection cost compared with only concerning key characteristics.

Consequently, this study initiates inspection planning by identifying inspection assignments consisting of geometric elements using GD&T information accompanied in the design. A GD&T information extraction system is created and a new taxonomy of tolerance is proposed here based on the definition of each tolerance characteristic. A data file of all identified inspection tasks with corresponding GD&T is eventually generated and stored in the database for subsequent steps.

4.2 Definition of tolerance taxonomy in the proposed system

A new taxonomy of tolerance is proposed in this study based on the definition of each tolerance characteristic in BS EN ISO 1101 and the related class diagram using UML is shown in Figure 4.1. It enables the system to process the design data regardless of the feature type by classifying all tolerance types with respect to three related basic features: line, surface and point. Additionally, all associated datums are decomposed into these basic features to maintain the consistency as shown in Figure 4.2.

Different characteristics of tolerance are no longer associated with the geometric classes such as form, orientation, location, and run-out. Instead, each characteristic is decomposed into geometric elements strictly according to the definition and instructions in the standard. Characteristics such as circular run-out, roundness and profile-any-line are only associated with curved line that belongs to element “line”. Similarly, profile-any-surface, total run-out, and cylindricity are associated with curved surface while total run-out is also associated with planar surface, both of which are related to the element “surface”. Some characteristics are non-exclusive to one type of geometric element such as parallelism, perpendicularity, and angularity, which can be associated with either “line” or “surface” depending on the actual applied tolerance. The same decomposition approach is implemented on datums existing in specified tolerances. Primary datum, secondary datum and tertiary datum are concerned here, which enable the system to comprehend different types of datum such as axis datum, surface extension datum, feature extension datum and target datum.

4.3 GD&T extraction and inspection tasks identification

As all tolerance characterizes are related with certain defined elements, it is then possible to identify the inspection tasks by analysing the resulting information. Unfortunately, there are existing issues in this process where the GD&T information is gathered.

Several researchers claimed that it is difficult to collect GD&T from CAD model due to the non-standardized storage methods that commercial CAD software use (Gao, 2006; Mohib et al., 2009). Most solid model formats such as IGES, VDA, CATPart and DXF present GD&T information in different data structures and some formats lack the ability to present the tolerance relationship to the part (Barreiro et al., 2003; Gao, 2006). It is then difficult to recall the design information in subsequent applications. Therefore, registration of GD&T information to the related feature is compulsory.

The proposed tolerance analysis module interactively collects the design data from metrology software by letting the user select the target tolerance. One major advantage of using the metrology software is the compatibility of comprehending different CAD formats, which overcomes the issue caused by non-standardization of GD&T data. Information such as tolerance characteristic, tolerance value and form of tolerance zone is accessed using specific software development kit and subsequently stored in the Inspection Planning System Database (IPSD). Using the taxonomy described earlier, those selected tolerances are decomposed and registered with basic feature ID and datum feature ID. The data structure specified in the database is described using UML class presentation, shown in Figure 4.3.

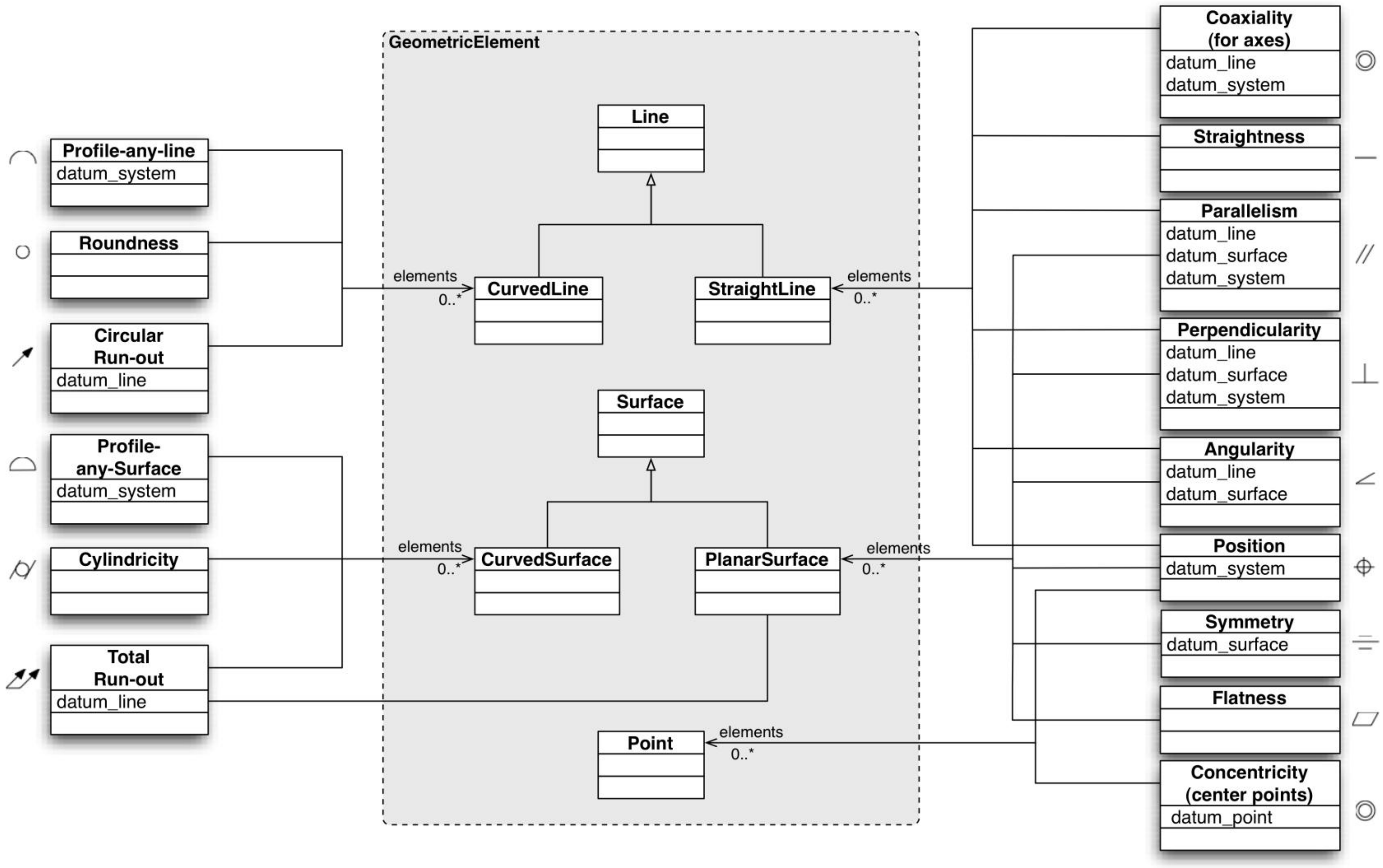


Figure 4.1 GD&T taxonomy presented in UML class graph

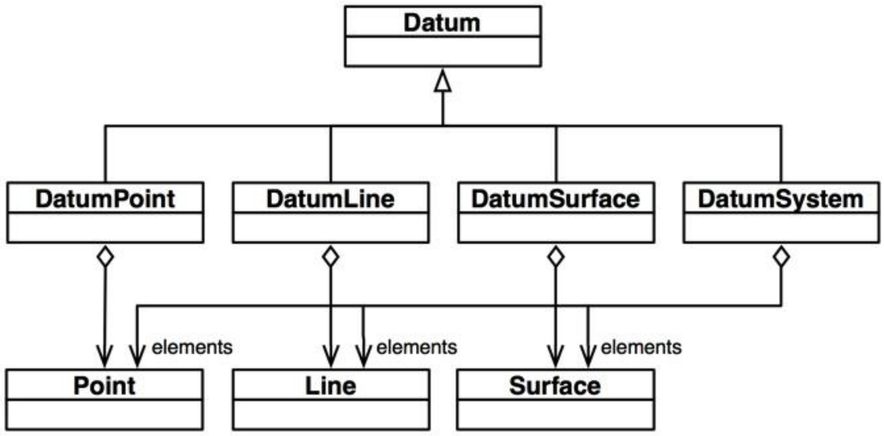


Figure 4.2 Datum classification in UML class graph

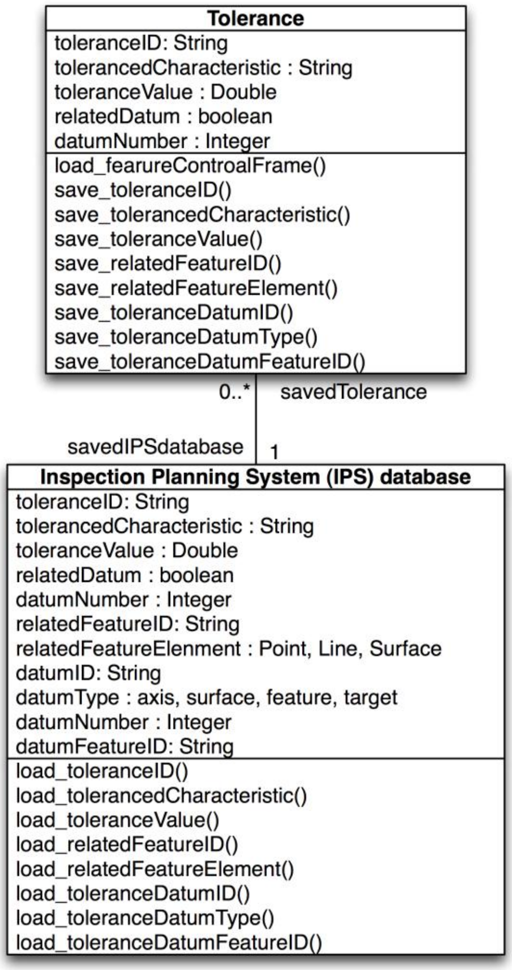


Figure 4.3 Data structure of the IPSD

Having gathered adequate information in the IPSD, the inspection tasks are accordingly identified. Every geometric element related with specified tolerance and datum can be retrieved by querying the database resulting in associated parameters such as feature ID and feature type. The involved feature element is then targeted as inspection task and stored in IPSD. One may notice that one basic feature can be utilized to define multiple tolerances and datums. Benefiting from the centralized database, the module is aware of this scenario and eliminates the redundancy in generating the inspection task.

The detailed data structure of the module is described in Figure 4.4 and the workflow is shown in Figure 4.5. Although the GD&T dominates the output of the module, it still allows the user to specify particular inspection tasks such as comparing the actual surface with the designed model where the deviation distribution over the surface can be generated.

An example of the inspection task identification is given below. A part is shown in Figure 4.6 with 4 tolerances, 1 dimension and 3 datums specified by the designer. After loading the database of metrology software, vital information regarding the GD&T is obtained and stored in IPSD. For instance, the parameters of datum A are retrieved including the datum type, associated object ID and associated surface ID, as shown in Figure 4.7. Tolerance information is obtained using the same strategy and as an example the surface profile tolerance assigned with respect to datum A, B and C is retrieved shown in Figure 4.8. Tolerance ID, tolerance type, associated object ID and surface ID, associated datums, and tolerance value are extracted. The inspection tasks with respect to the above two examples are generated accordingly in IPSD, presented in Table 4.1.

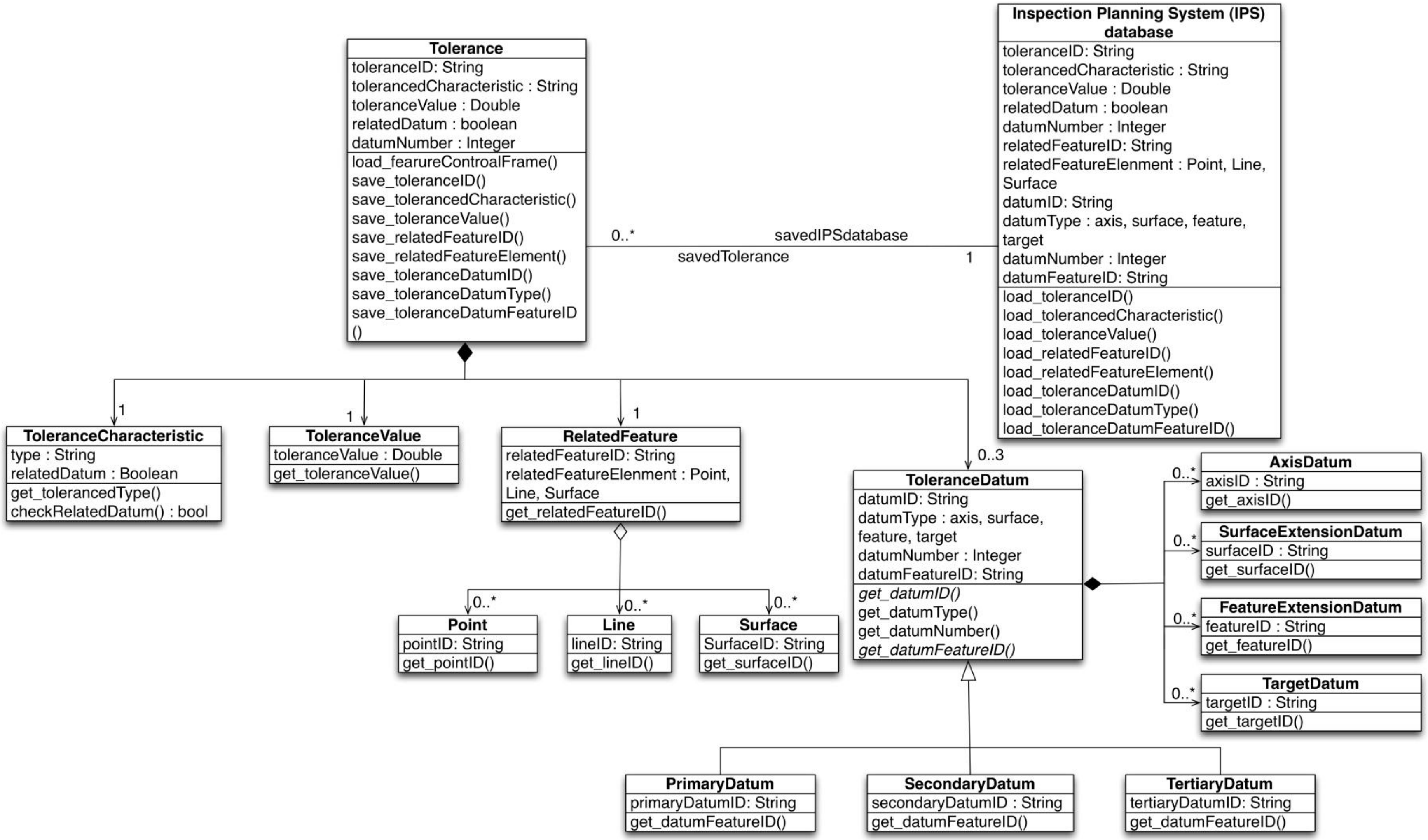


Figure 4.4 Structure of the tolerance analysis module in UML

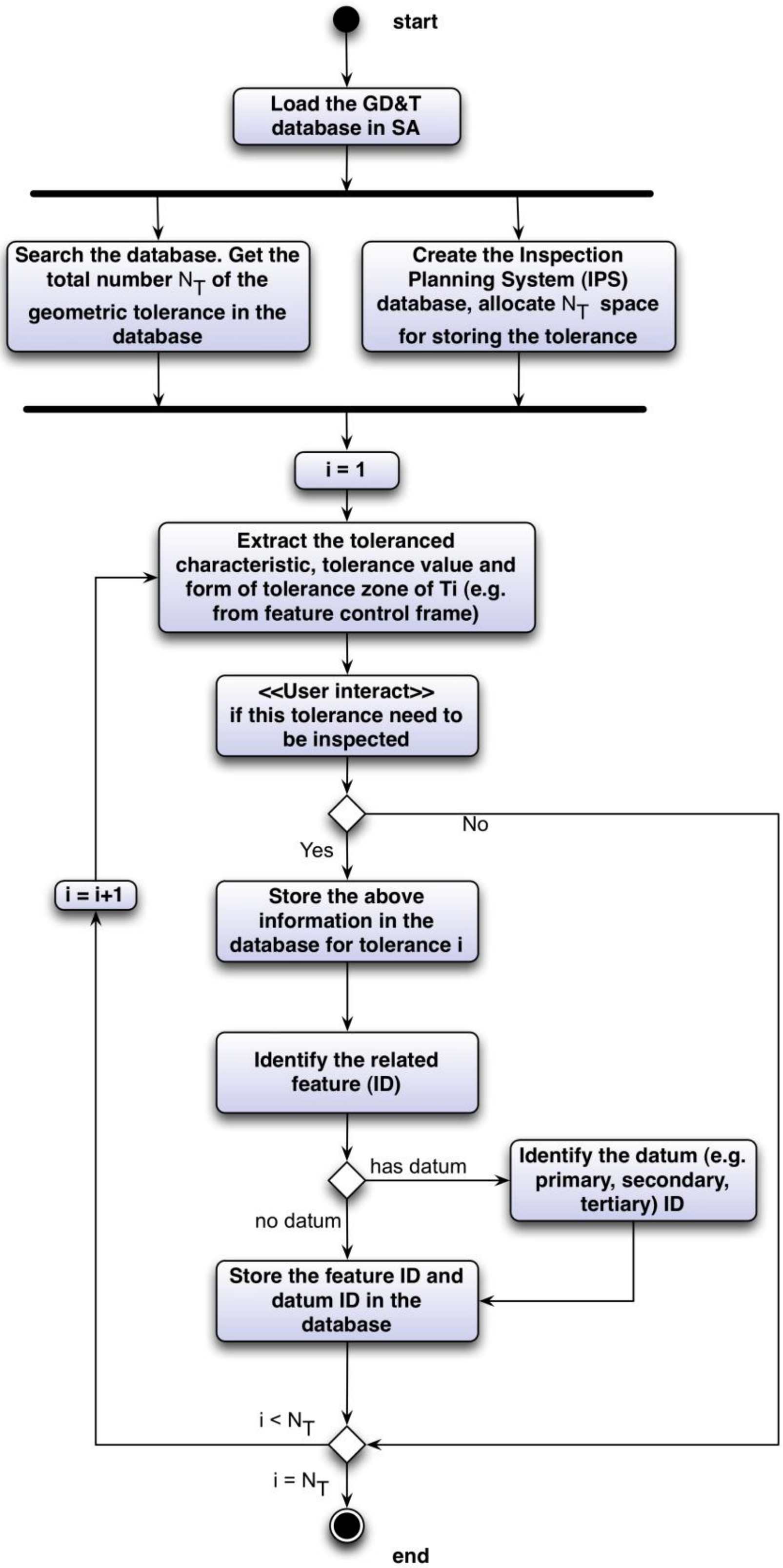


Figure 4.5 Workflow of the tolerance analysis

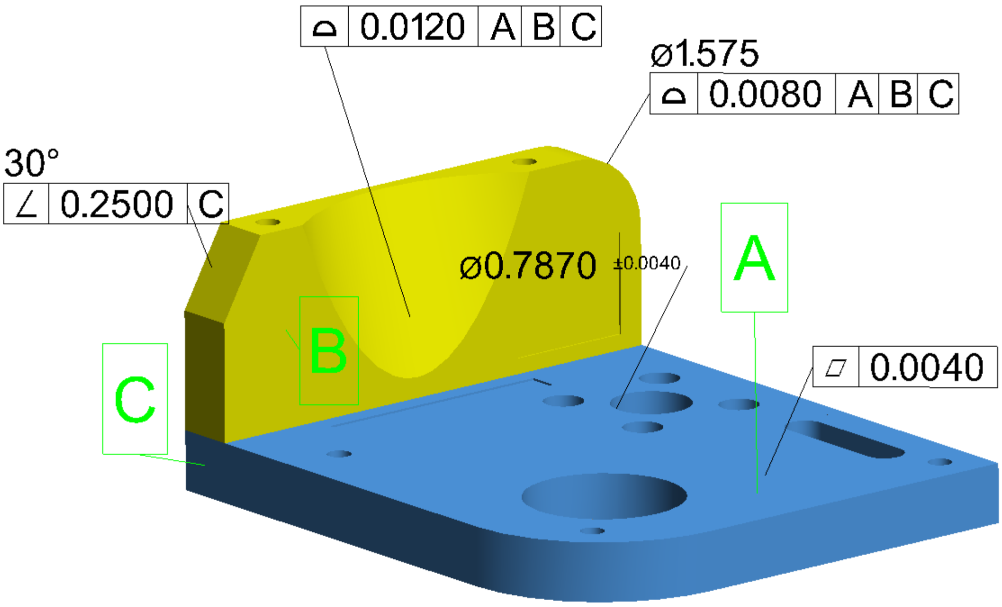


Figure 4.6 An example of inspection tasks identification by tolerance analysis

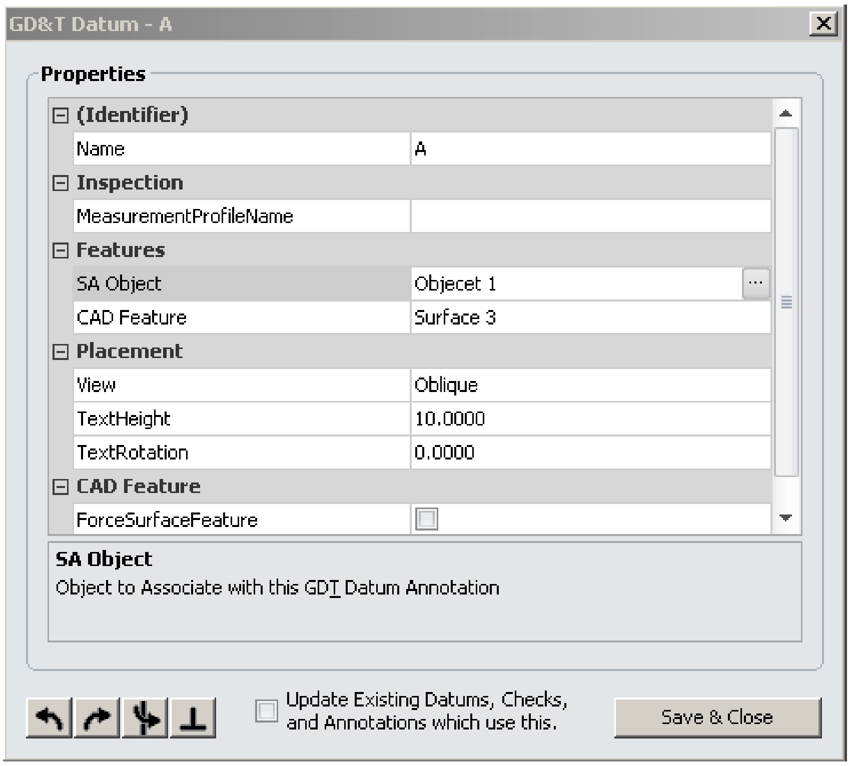


Figure 4.7 The retrieve of the parameters of datum A

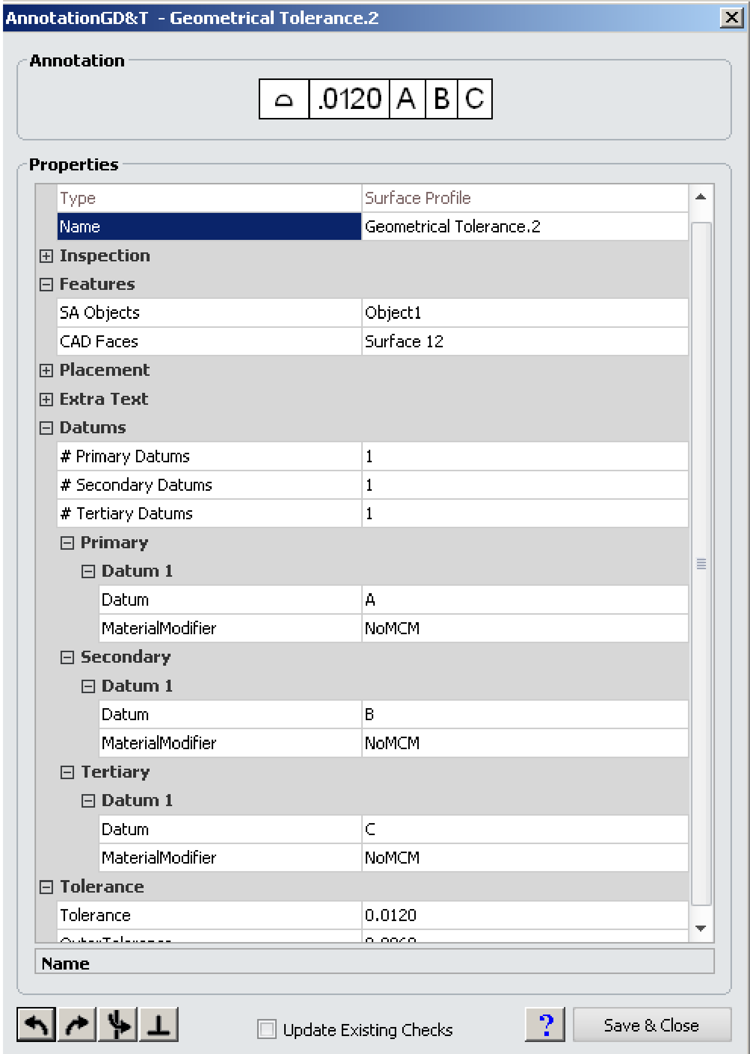


Figure 4.8 The retrieval of parameters of surface profile tolerance

Table 4.1 Identified inspection tasks from tolerance 2

Task_ID	1	2
Feature_relatedObject	object_1	object_1
Feature_elementType	surface	surface
Feature_ID	surface_12	surface_3
Related_Tolerance	GeometricalTolerance.2	Geometrical Tolerance.2 Geometrical Tolerance.3
Related_Datum		A

4.4 Summary

Aiming to determine the inspection tasks at the initial stage of planning process, this chapter discusses the drawbacks and unsuitability of current techniques such as feature recognition and extraction. Inspection assignments identification based on geometric dimensioning and tolerancing is thus presented with a new taxonomy of tolerance classification that facilities the process. The structure of the data management is also described with a demonstration that reveals the effectiveness of the approach.

CHAPTER 5 INSTRUMENT SELECTION AND INSPECTION POINT DETERMINATION

5.1 Introduction

Instrument selection is deemed as a compulsory and critical process in inspection planning for large volume metrology applications. The process identifies capable and suitable metrology devices with respect to the desired measurement tasks. Most research efforts in the past have focused on probe selection for CMMs (Beg and Shunmugam, 2002; Mohib et al., 2009; Zhao et al., 2009). However, increasing demand for accurate measurement in large scale and complex assembly and fabrication industries, such as aerospace and power generation makes these industries need to invest in different measurement systems and technologies. The increasing number of systems with different capabilities create difficulties in selecting the most competent large volume metrology (LVM) instrument for a given measurement task. Research in this area is sketchy due to having vast candidates of qualified instruments and at the same time the complexity of understanding their real capabilities.

Traditional inspection planning carried out for CMMs chooses the most suitable probe manually among three to five probes, based on pre-defined criteria such as the accuracy requirement of the measurement and accessibility limitation of the setup (Ziemian and Medeiros, 1998; Limaiem and ElMaraghy, 1999; Zhang et al., 2000; Cho et al., 2004; Cho et al., 2005; Wong et al., 2006; Mohib et al., 2009). Unlike probe selection in most IPP system for coordinate measuring machines, selecting suitable LVM instruments faces more complexity and vagueness due to the large number of available instruments and uncertain relationships among instrument performance criteria. Previous work (Cai et al., 2008; Cai et al., 2010; Muelaner et al., 2010) has successfully defined the process of measurability analysis. In this process a variety of criteria are specified with corresponding evaluation methods. Instrument selection is based on the result of measurability analysis although automation is severely limited. However, many task requirements and related importance, which is usually unequal, are ambiguous while defining the criteria. In addition, some parameters of alternative instruments cannot be quantified at this stage without detailed sampling strategy and instrument configuration e.g. inspection time and inspection cost. Vague relationship among criteria also leads to uncertain decision, such as the trade-off relationship between measuring uncertainty and process tolerance with specified tolerance interval, which implies relationship between cost and uncertainty (Flack and Hannaford, 2005). Figure 5.1 illustrates three measurement instruments with different uncertainties. Machine A has the largest measuring uncertainty with lowest cost for purchasing and utilizing, but requires tightest uncertainty for processing. By contrast, Machine C obtains a relaxed process tolerance interval by decreasing measuring uncertainty significantly.

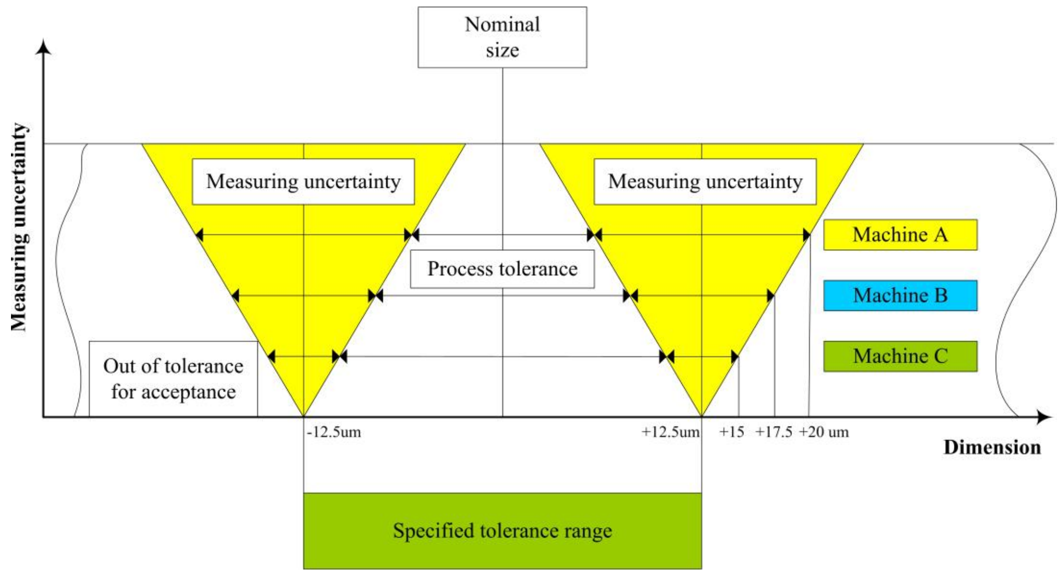


Figure 5.1 Trade-off relationship between measuring uncertainty and process tolerance (Flack and Hannaford, 2005)

Decision is ambiguous while attempting to use more accurate instrument that has a potentially higher cost. Moreover, in most applications, the selection process involves more than one decision maker (DM) e.g. designers and metrologists. The assigned preference of alternatives may be different due to the unique understanding of the task and unequal knowledge of the instruments. This leads to different assigned weights when the significance of different criteria is evaluated by DMs. It is therefore formulated as a multi-criteria multi-person decision making problem.

Fuzzy set theory (FST) was first introduced by Zadeh (1965) with the objective of denoting vagueness and fuzziness in a set and processing unquantifiable and incomplete information in decision problems. Fuzzy linguistic models enable the conversion of vague verbal expressions such as ‘extremely’, ‘very’ and ‘medium’ into fuzzy numbers, which allows DMs to estimate the performance of alternatives and make decision based on quantitative data. Atanassov (1986) defined the concept of intuitionistic fuzzy set (IFS) as a generalization of FST, characterized by a membership function and a non-membership function. IFS with technique for order performance by similarity to ideal solution (TOPSIS) have recently attracted great attention in multi-attributes decision-making (MADM) process due to the consideration of both positive-ideal and negative-ideal solutions (Karsak, 2002; Bozdağ et al., 2003; Chen et al., 2006; Boran et al., 2009; Onuut et al., 2009). The method has been successfully applied in different regions such as robot selection and supply chain management (Karsak, 2002; Chan et al., 2008; Boran et al., 2009). By using fuzzy group decision-making, not only vague criteria such as inspection cost, time and speed can be taken into account, but also other important crisp criteria given by different weighted decision makers are considered. In the meantime, precise decisions can be made while conflicting criteria are assessed using different weights.

A two-phased instrument selection system is proposed in this chapter to solve this decision making problem using fuzzy set theory and a numerical case study is given to demonstrate the effectiveness of the system.

5.2 Proposed instrument selection process

5.2.1 Measurability characteristics

Over the last decade, the concept and methodologies of Quality Characteristics (QCs) have been studied and practiced in many world-class companies, and QCs play a significant role in product lifecycle management (PLM) and in collaborative and global product development (Chin et al., 2003). There are different levels of attributes associated with QCs including basic attributes, lifecycle attributes, interrelation attributes and measurement attributes, which are all utilized to perform global planning and resource allocation (Dai and Tang, 2008). In order to borrow this planning and optimization approach to specific measurement aims, measurability characteristics (MCs) are proposed with associated attributes such as volumetric coverage of the measurement, environmental conditions, material properties, inspection cost, accuracy, speed and Technology Readiness Level (TRL). Analysing the attributes of MCs facilitates the classification of measurement aims and measurement instruments by mapping different MCs to the appropriate measurement process. The proposed MCs are categorized into two groups to be assessed in two phases, respectively.

5.2.1.1 Crisp measurability characteristics

Crisp MCs are defined as C_{ci}, which can be precisely assessed based on the following criteria:

1) The environmental conditions under which the inspection task will be carried out. For instance the temperature, altitude and humidity must meet the instrument specified capabilities. One issue that should be addressed is that an uncertainty budget describing the uncertainty components is mandatory for any traceable measurement which not only contains the uncertainty of the instrument but also takes into account other effects that degrade the measurement accuracy e.g. environmental conditions, thermal expansion and SMR errors for laser trackers (B89.4.19, 2006). Both temperature variations in time and temperature gradients along the measurement volume will affect the suitability and capability of laser-based systems (Puttock, 1978; Muelaner et al., 2008). Other environmental variables, such as humidity and barometric pressure, also have an effect on measurements. The consideration of environmental factors should eliminate instruments whose operating limits fall outside of the expected environment.

2) The inspection range or the distance of measurement points from the instrument. It is difficult to obtain precise information on the scale of the inspection task at this stage. LVM instruments feature the ability of mobilizing around the target and therefore the volumetric coverage of the specific instrument generally will not confine the applicability. Nevertheless, other MCs such as inspection speed and accuracy are compromised if the instrument is relocated during the inspection.

3) The material properties of the target product. For instance, material with low reflectivity coefficient and transparent surfaces cannot be measured accurately by instrument employing laser based measurement system. For aluminum or plastics, magnetic targets cannot be applied, which are often used with photogrammetry systems (Cuypers et al., 2009).

4) The stiffness of the product, e.g. only non-contact system can be deployed on product with high flexibility due to undesired surface movements.

5) The uncertainty requirement of the inspection, e.g. the uncertainty of the selected instrument should be confined by decision rules (BS EN ISO 14253-1, 1999; B89.7.3.1, 2001)

Among those crisp MCs, uncertainty capability is vital for any measurement instrument since without knowledge of the uncertainty of the instrument used to take a measurement it is not possible to draw any conclusions about part conformance from those measurements. Furthermore even where the uncertainty of the measurement instrument is known, if it represents a significant proportion of the part tolerance there will be frequent occurrences of measurements where it is not possible to state either conformance or non-conformance. As shown in Figure 5.2, although inspection results from instruments A and D are within the specified tolerance zone, the decision of acceptance or rejection can not be made due to the uncertainty bands that exceed the tolerance zone. Similarly, result from instrument C cannot be rejected with confidence as the uncertainty band partially falls into the tolerance zone.

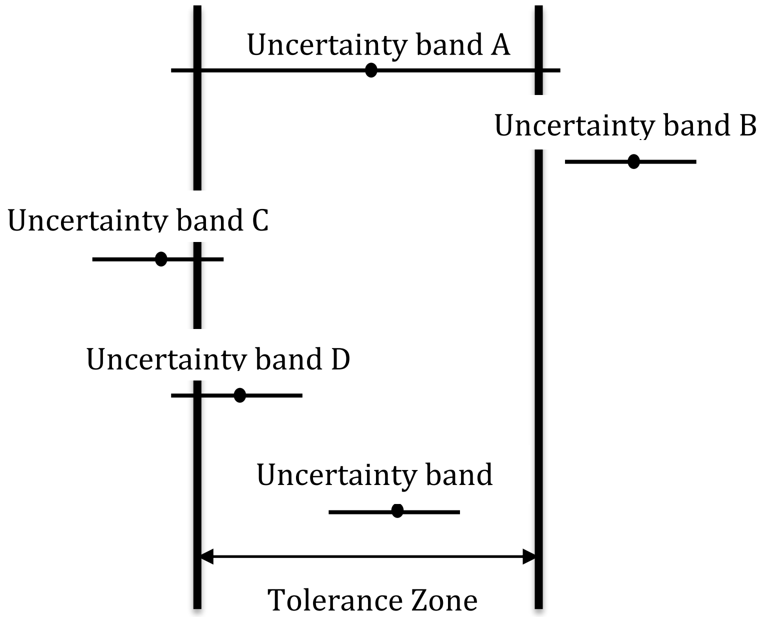


Figure 5.2 An example of decision rules for part inspection

The uncertainty of a measurement must be added to the lower tolerance band to give a minimum acceptance value. Similarly the uncertainty must be subtracted from the upper tolerance band to give a maximum acceptance value. When the part is measured the reading must be within the range of the acceptance values in order to prove conformance (BS EN ISO 14253-1, 1999). This range of acceptance values, or residue tolerance, is the tolerance required by the manufacturing process. Most attempts to consider uncertainty only take the uncertainty of measured points into account while integrating multi-instruments for enhanced results in terms of reduced uncertainty and measurement time (Mohib et al., 2009). An approach for assessing the accuracy against each particular measurement task is absent from the literature. A requirement therefore exists for a measurement uncertainty capability index to ensure the measuring equipment and measurement processes are suitable and capable of achieving product quality objectives.

The measurement accuracy capability index has been defined in several ways in order to ensure that the measuring equipment and measurement process are suitable and capable of achieving product quality objectives (Kurekova, 2001). It has been represented as a criterion to evaluate the performance of laser-based spherical coordinate measurement systems in the large volume metrology region (B89.4.19, 2006). However, previous definitions of this index are all based on the results after measurement has been conducted. According to the requirements of measurability analysis, it is imperative to introduce an index which is capable of indicating the instrument's measurement accuracy capability before any measurement has been conducted.

As described in ASME B89.7.3.1 (2001), simple acceptance and rejection using an n:1 rule is the most common form utilized in industry and is the descendant of MIL-STD 45662A (1988). The product is acceptable if the measurement result lies within the specification zone and otherwise is rejected, provided that the magnitude of the measurement uncertainty interval is no larger than the fraction 1/N of the specification zone. Recently, since engineering tolerances have been reduced significantly, a new four-to-one ratio rule has replaced the well-used ten-to-one ratio (MIL-STD 45662A, 1988). As shown in Figure 5.3, the measurement product will be acceptable if the uncertainty interval associated with the measurement results is no lager than the one-fourth of the engineering specification zone. This requires the expanded uncertainty to be no lager than the one-eighth of the specification zone. This four-to-one ratio rule can be formulated as:

$$\frac{USL-LSL}{2U} = \frac{T}{2U} \geq 4 \qquad \text{(Eq5.1)}$$

where tolerance interval T is the difference between the upper specification limit (USL) and the lower specification limit (LSL) and U is the $k=2$ expanded uncertainty associated with the measurement results.

Consequently, the measurement accuracy capability index is defined as:

$$C_m = \frac{USL-LSL}{U} = \frac{T}{U} \qquad \text{(Eq5.2)}$$

where U is the expanded uncertainty *($k=2$)* of the corresponding instrument.

This equation reveals that an instrument is capable of conducting an inspection, which has a tolerance interval T, if $C_m \geq 8$ according to Eq5.1. This index implies that the measurement accuracy capability of an instrument is proportional to C_m.

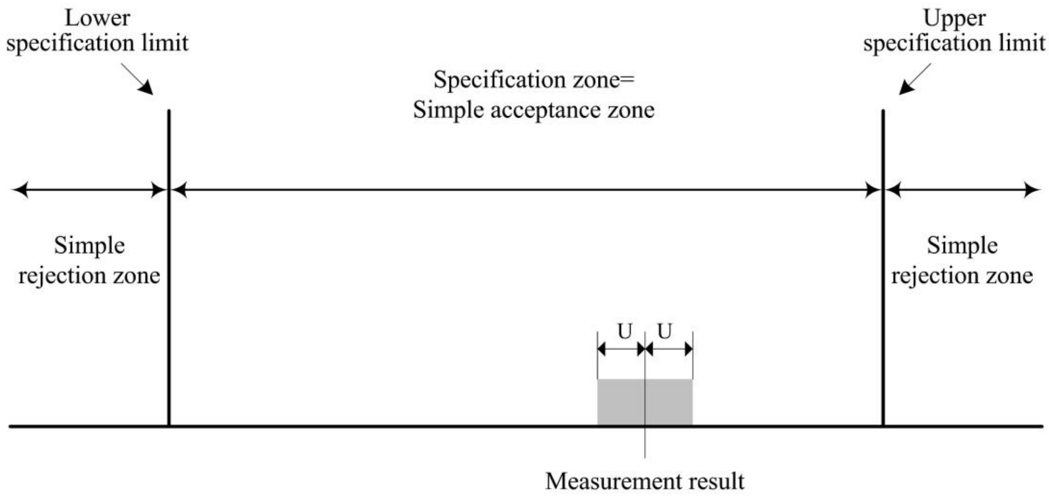


Figure 5.3 An example of simple acceptance and rejection using a 4:1 ratio (ASME, 2001).

As stated in ASME B89.4.19, a laser tracker quantifies its expanded uncertainty *(k=2)* by the Maximum Permissible Error (MPE). Hence, all suppliers of laser trackers who intend to verify their products by this standard should provide the formula of MPE against distance from laser tracker to the target. MPE is deemed to be a generic parameter introduced by a standard instead of the expanded uncertainty of a laser tracker. Moreover, MPE takes into consideration the effects on uncertainty from all possible sources rather than merely the instrument. Consequently, the measurement accuracy index for a laser tracker can be represented as:

$$C_m = \frac{USL-LSL}{MPE} = \frac{T}{MPE} \quad \text{(Eq5.3)}$$

It is therefore desirable to obtain the MPE for every instrument where applicable.

5.2.1.2 Fuzzy measurability characteristics

It is beneficial to define those fuzzy MCs due to the incomplete information at this early planning stage and conflicting relationship among them. For instance, uncertainty performance of most instruments is related to the measuring distance to the target, which is unknown without the detailed configuration and topological plan of a specific instrument. Measurement speed and cost can only be determined when both sampling strategy and system setup are available. In addition to that, a non-linear trade-off relationship exists between cost and uncertainty resulting in ambiguous decision. An attempt to use more accurate instrument has a potentially higher cost. Under this circumstance, criteria with vagueness are defined as C_{fi}:

1) The uncertainty capability of the chosen instrument. Although the uncertainty capability requirement of the tasks has been satisfied by assessing the crisp MC, the vague uncertainty capability allows the system to take the performance of capable instruments into account when the decision of selecting the most suitable instrument is made.

2) Overall cost of deploying the instrument which includes recurring cost e.g. purchasing the system and mandatory training, and non-recurring cost e.g. maintenance, depreciation.

3) Measurement speed.

4) Technology Readiness Level (TRL) of the instrument. Consulting the most common definitions of TRL published by the Department of Defence (2006) and the National Aeronautics and Space Administration (Mankins, 1995) the TRL for large volume measurement technologies is composed of four generic levels that classify all measurement principles and instruments shown in Table 5.1.

5.2.2 Proposed system

System structure is presented using a UML class diagram in Figure 5.4. MCs are identified and grouped into quantitative and qualitative attributes. Phase-1 enables the filtration of instrument based on crisp requirements of the inspection task. The remaining instruments are assessed in Phase-2 according to qualitative criteria and a rank list of alternatives is given as the result.

Table 5.1 Proposed TRL levels

Technology Readiness Level	**Description**
Level 1 Basic measurement principles observed and reported	Lowest level of technology maturity. At this level, scientific research of the measurement principles starts to be translated into applied research and development.
Level 2 Measurement system or subsystem model or prototype demonstration	Practical demonstration of the measurement principle using a representative model or prototype system must be carried out in order to demonstrate the fidelity of the measurement technology either in a company or the laboratory.
Level 3 Actual system completed and sold in the commercial market	Entire measurement system must be supplied with essential peripherals and support devices as well as adequate operation and control software.
Level 4 Actual system qualified by international standard	Verification approach for measurement accuracy capability, stability and reliability must be demonstrated and standardised by a National Measurement international standard.

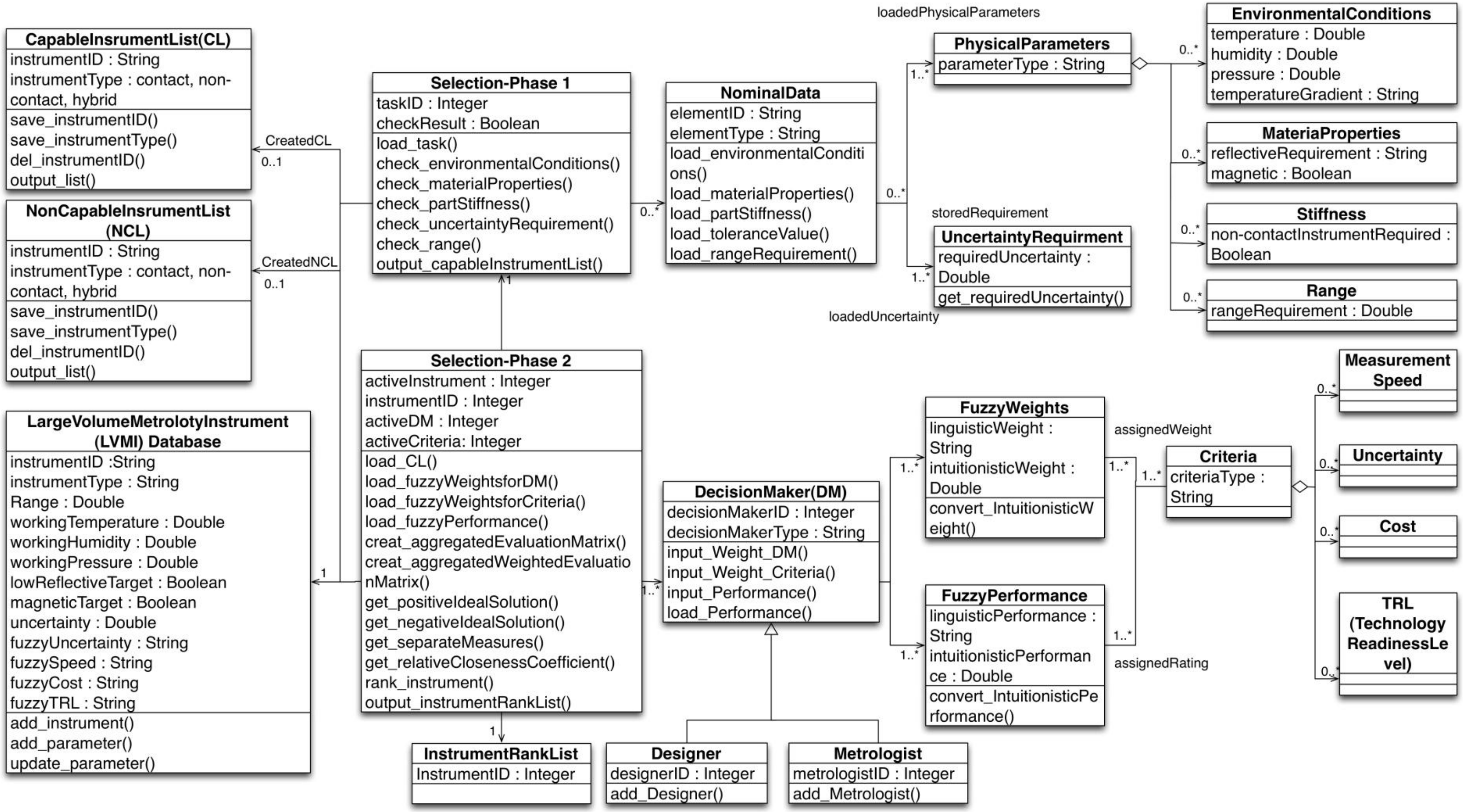


Figure 5.4 System structure of instrument selection in UML

5.2.2.1 Phase 1: instrument filtration

In this phase, inspection features previously stored in IPSD are retrieved individually with associated parameters. Quantitative requirements defined by inspection tasks are assessed sequentially. This includes measurement volume, environmental conditions, material properties and required uncertainty level. Each of the assessment serves the system as a filter, searching through the LVM Instrument Database and eliminating the instrument that is not suitable for the task. After several filtering loops, applicable alternatives are obtained and stored in IPSD for re-assessing in the next phase. Figure 5.5 shows the algorithm of Phase 1 using UML activity diagram. The following steps detail the algorithm of instrument filtration.

Step 1 Retrieving inspection requirements.

In this step, inspection features extracted from design are retrieved individually with associated parameters. Crisp MCs are then obtained accordingly and set as criteria $C_{c\mathrm{i}}$ for later evaluation. Table 5.2 shows an example of interpreted crisp MCs.

Step 2 Filtering the instruments.

A capable instrument list (CPL) and an incapable instrument list (IIL) are created to temporarily store the result, facilitating the filtration process. Instruments located in the large volume metrology instrument database are activated sequentially with associated specification I_{si}. The data structure of the database shown in Figure 5.4 and Table 5.3 are given as an example of FARO laser tracker. Comparisons are then carried out between C_{ci} and I_{si} in such order: stiffness of the product, environmental conditions, material properties, uncertainty requirement, and inspection range. By assessing the more obvious criteria first the sequence ensures that minimum comparing loops are employed. Once unsatisfied criterion is detected, I_i is removed from CIL to IIL and the rest of C_{ci} are cancelled to save computational power.

The output of Phase 1 is a list with all capable instruments with respect to inspection tasks and it is passed to the next stage for further selection.

Table 5.2 Example of crisp MCs

Inspection ID		**1**
Crisp MCs		**Details**
Environmental conditions	Temperature	25°
	Altitude	500 m
	Humidity	35%
Stiffness limitation		contact& non-contact
Material property		magnet applicable
Uncertainty requirement		0.2 mm
Range		14m

Table 5.3 Stored data of FARO Laser Tracker

Instrument ID	**1**
Instrument Type	Laser Tracker
Maximum Operating Temperature	50°
Minimum Operating Temperature	-15°
Maximum Operating Altitude	2450 m
Minimum Operating Attitude	-700 m
Maximum Acceptable Humidity	95% non-condensing
Minimum Acceptable Humidity	0
Low reflective target	no
Magnetic target	yes
Range	55 m
Uncertainty	ADM: 16µm + 0.8µm/m Interferometer: 4µm + 0.8µm/m

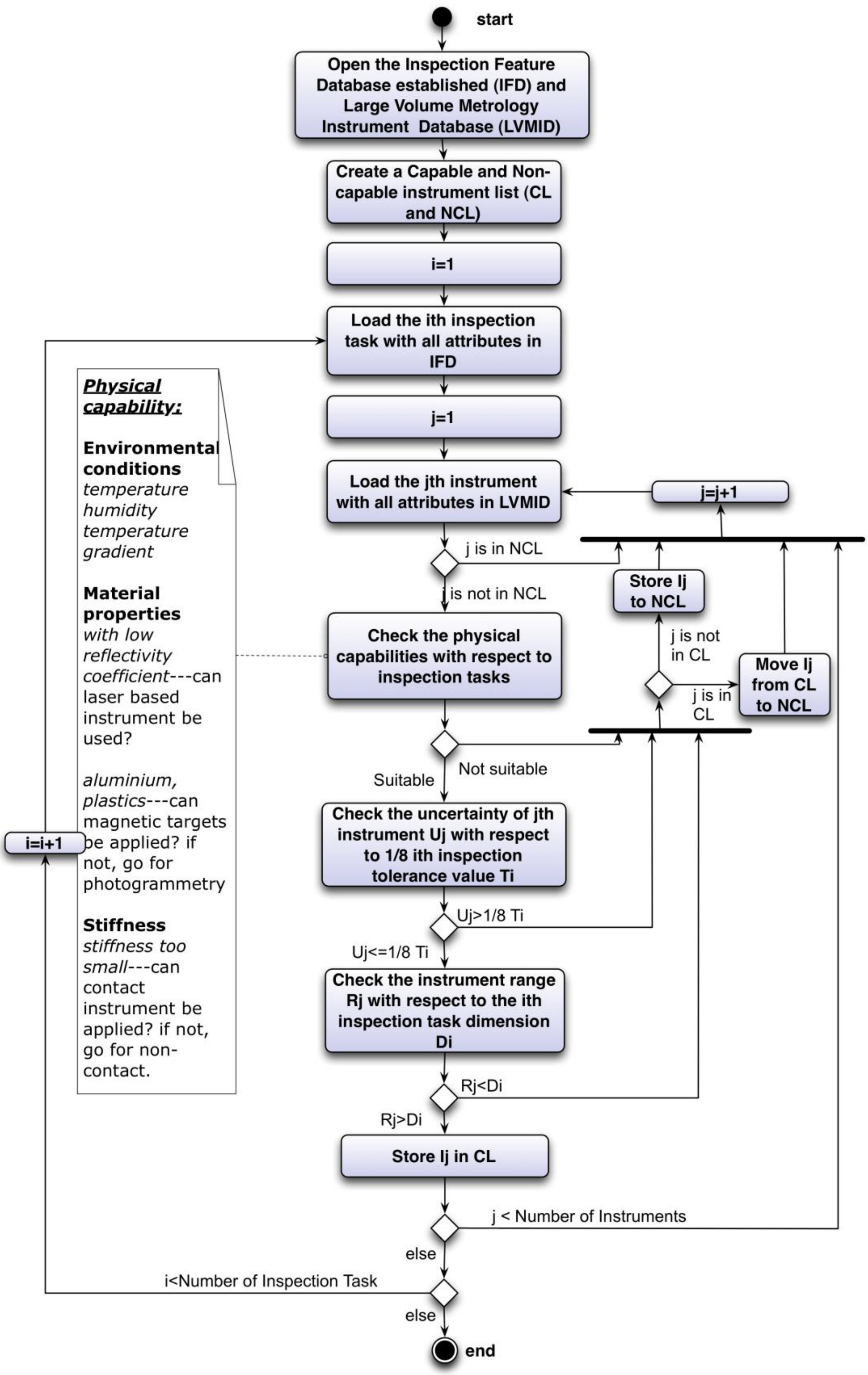


Figure 5.5 UML activity diagram of instrument selection-Phase 1

5.2.2.2 Phase 2: fuzzy instrument selection

Zadeh (1965) defined a fuzzy set A as:

$$A = \{\langle x, \mu_A(x)\rangle | x \in X\} \quad \text{(Eq5.4)}$$

where $\mu_A(x): X \rightarrow [0,1]$ is the membership function indicating the degree that element x belongs to the set A. The closer the value of $\mu_A(x)$ is to 1, the more x belongs to A.

Atanassov (1986) first introduced the intuitionistic fuzzy set (IFS) as an extension of the classical FST. Characterized by both membership function and non-membership function, IFS has been particularly preferred in decision-making applications. Recently, the technique for order performance by similarity to ideal solution (TOPSIS) plays an important role in multi-attribute decision-making (MADM) process (Chen et al., 2006; Onuut et al., 2009). It takes both positive-ideal and negative-ideal solution into account, ensuring more precise decision when different assessing perspectives and trade-off relations exist among decision makers and criteria.

Intuitionistic fuzzy set A can be written as:

$$\mathrm{A} = \{\langle \mathrm{x}, \mu_\mathrm{A}(\mathrm{x}), \nu_\mathrm{A}(\mathrm{x})\rangle | \mathrm{x} \in \mathrm{X}\} \quad \text{(Eq5.5)}$$

where $\mu_A(x): X \rightarrow [0,1]$ is the membership function and $\nu_A(x): X \rightarrow [0,1]$ is the non-membership function with the condition that

$$0 \leq \mu_A(x) + \nu_A(x) \leq 1 \quad \text{(Eq5.6)}$$

Another unique parameter $\pi_A(x)$ known as the intuitionistic fuzzy index is defined as:

$$\pi_\mathrm{A}(\mathrm{x}) = 1 - \mu_\mathrm{A}(\mathrm{x}) - \nu_\mathrm{A}(\mathrm{x}) \quad \text{(Eq5.7)}$$

The multiplication operator of two IFSs A and B in a finite set X is defined as

$$\mathrm{A} \otimes \mathrm{B} = \{\mu_\mathrm{A}(\mathrm{x}) \cdot \mu_\mathrm{B}(\mathrm{x}), \nu_\mathrm{A}(\mathrm{x}) + \nu_\mathrm{B}(\mathrm{x}) - \nu_\mathrm{A}(\mathrm{x}) \cdot \nu_\mathrm{B}(\mathrm{x}) | \mathrm{x} \in \mathrm{X}\} \quad \text{(Eq5.8)}$$

Boran et al. (2009) and Onut et al. (2009) proposed similar approaches to solve the MADM supplier selection problem using TOPSIS. IFSs were utilized to select the appropriate supplier by aggregating individual opinions of DMs for weighting the importance of both criteria and alternatives (Boran et al., 2009). Their research results demonstrated the effectiveness of the approach. A similar method is adopted in this work. The following steps detail the algorithm and process of applying IFSs to instrument selection.

Step 1 Modelling the MADM problem.

(a) Let the capable alternative instruments stored in CIL from Phase 1 be a finite set $I = \{I_1, I_2, \dots, I_n\}$.

(b) Let the fuzzy MCs be a finite criteria set $C = \{C_1, C_2, \dots, C_m\}$, which includes instrument uncertainty, overall cost, inspection speed and TRL.

(c) Let $D = \{D_1, D_2, \dots, D_l\}$ be the decision maker set including both designers and metrologists in the decision making process.

(d) Let $P^{(k)} = (p_{ij}^{(k)})_{m \times n}$ denote the $m \times n$ decision matrix of kth decision maker, where p_{ij} is the performance rating of alternative instrument I_i with respect to criterionC_j.

(e) Alternative instruments are linguistically rated by DMs using terms defined in Table 5.4. The importance of DMs is evaluated using the linguistic term in Table 5.5, where the typical converged IFSs are also given. Table 5.6 is used to collect the importance of different DMs as well as their individual ratings on different weighted criteria.

Table 5.4 Linguistic performance and IFNs

Linguistic Performance Evaluation	IFNs
Extremely good (EG)/extremely high (EH)	(1.00,0.00)
Very good (VG)/very high (VH)	(0.80,0.10)
Good (G)/high (H)	(0.70,0.20)
Fair (F)/medium (M)	(0.50,0.40)
Bad (B)/low (L)	(0.25,0.60)

Table 5.5 Linguistic importance and IFNs

Linguistic Importance	IFNs
Very Important	(0.90,0.10)
Important	(0.75,0.20)
Medium	(0.50,0.45)
Unimportant	(0.35,0.60)
Very Unimportant	(0.10,0.90)

Table 5.6 Instrument performance rating

Weights			ω_{DC1}	ω_{DC2}	ω_{DC3}
	Criteria	Instrument	DM 1	DM 2	DM 3
ω_{C1}	Uncertainty	I_1			
		I_2			
		I_3			
ω_{C2}	Measurement Speed	I_1			
		I_2			
		I_3			
ω_{C3}	Cost	I_1			
		I_2			
		I_3			
ω_{C4}	TRL	I_1			
		I_2			
		I_3			

Step 2 Assigning linguistic importance to designers and metrologists, and calculating the corresponding weights.

Let $W_{D_k} = [\mu_k, \nu_k, \pi_k]$ be the intuitionistic fuzzy rating of kth decision maker using linguistic term and the weight of kth decision maker is calculated as:

$$\omega_k = \frac{\left(\frac{\mu_k}{\mu_k+\nu_k}\right)}{\sum_{k=1}^{l}\left(\mu_k+\pi_k\left(\frac{\mu_k}{\mu_k+\nu_k}\right)\right)} \tag{Eq5.9}$$

where $0 \leq \omega_k \leq 1$ and $\sum_{k=1}^{l} \omega_k = 1$.

Step 3 Aggregating the decision matrix with respect to the individual performance rating of decision makers.

Having fused individual opinion $P^{(k)}$ from all weighted DMs, group opinion is aggregated as the intuitionistic fuzzy decision matrix. IFWA operator is utilized in the aggregation process proposed by Xu (2007a; 2007b)

$$\mathrm{p_{ij}} = \mathrm{IFWA_\omega}\left(\mathrm{p_{ij}^{(1)}}, \mathrm{p_{ij}^{(2)}}, \dots, \mathrm{p_{ij}^{(l)}}\right)$$

$$= [1 - \prod_{k=1}^{l}\left(1 - \mu_{ij}^{(k)}\right)^{\omega_k}, \prod_{k=1}^{l}\left(\nu_{ij}^{(k)}\right)^{\omega_k},$$

$$\prod_{k=1}^{l}\left(1 - \mu_{ij}^{(k)}\right)^{\omega_k} - \prod_{k=1}^{l}\left(\nu_{ij}^{(k)}\right)^{\omega_k}] \tag{Eq5.10}$$

The matrix is then written as

$$P = \begin{bmatrix} p_{11} p_{12} & \cdots & p_{1m} \\ p_{21} p_{22} & \cdots & p_{2m} \\ \vdots \quad \vdots & \ddots & \vdots \\ p_{n1} p_{n2} & \cdots & p_{nm} \end{bmatrix} \tag{Eq5.11}$$

where $p_{ij} = (\mu_{I_i}(x_j), \nu_{I_i}(x_j), \pi_{I_i}(x_j))$.

Step 4 Assigning linguistic importance to the criteria and calculating the corresponding weights.

The system allows decision makers to assign different weights to each criterion, which is a key advantage for emphasizing the vague relation existing among criteria, e.g. uncertainty, inspection speed and cost.

It is assumed that the kth decision maker weights the jth criterion with an intuitionistic number ${w_j}^{(k)} = \left[\mu_j^{(k)}, \nu_j^{(k)}, \pi_j^{(k)}\right]$. The overall weight of the jth criterion is calculated using IFWA operator:

$$w_j = IFWA_\lambda\left(w_j^{(1)}, w_j^{(2)}, \dots, w_j^{(l)}\right)$$

$$= [1 - \prod_{k=1}^{l}\left(1 - \mu_{ij}^{(k)}\right)^{\omega_k}, \prod_{k=1}^{l}\left(\nu_{ij}^{(k)}\right)^{\omega_k},$$

$$\prod_{k=1}^{l}\left(1 - \mu_{ij}^{(k)}\right)^{\omega_k} - \prod_{k=1}^{l}\left(\nu_{ij}^{(k)}\right)^{\omega_k}] \quad \text{(Eq5.12)}$$

and the weight matrix is then formed as

$$W = [w_1, w_2, \dots, w_j] \quad \text{(Eq5.13)}$$

where $w_j = (\mu_j, \nu_j, \pi_j)$.

Step 5 Creating the weighted decision matrix by aggregating *P* and *W*.

P and *W* are multiplied using Eq5.8 resulting in the weighted intuitionistic fuzzy decision matrix:

$$P \otimes W = \{\langle x, \mu_{I_i}(x) \cdot \mu_W(x), \nu_{I_i}(x) + \nu_W(x) - \nu_{I_i}(x) \cdot \nu_W(x)\rangle \middle| x \in X\} \quad \text{(Eq5.14)}$$

The matrix is then written as

$$P' = \begin{bmatrix} r'_{11} r'_{12} & \cdots & r'_{1m} \\ r'_{21} r'_{22} & \cdots & r'_{2m} \\ \vdots \quad \vdots & \ddots & \vdots \\ r'_{n1} r'_{n2} & \cdots & r'_{nm} \end{bmatrix} \quad \text{(Eq5.15)}$$

and

$$\pi_{I_iW}(x_j) = 1 - \mu_{I_i}(x) \cdot \mu_W(x) - \nu_{I_i}(x) - \nu_W(x) + \nu_{I_i}(x) \cdot \nu_W(x) \quad \text{(Eq5.16)}$$

where $p'_{ij} = (\mu_{I_iW}(x_j), \nu_{I_iW}(x_j), \pi_{I_iW}(x_j))$.

Step 6 Calculating the separation distance of each alternative to positive-ideal solution and negative-ideal solution.

Criteria such as uncertainty, TRL and speed denoted by C_B are beneficial while rating the alternative instruments. By contrast, the overall cost is considered as cost criterion denoted by C_C. The intuitionistic fuzzy positive-ideal solution I^* and negative-ideal solution I^- are defined as:

$$\mathrm{I}^* = (\mu_{\mathrm{I}^*\mathrm{W}}(\mathrm{x_j}), \nu_{\mathrm{I}^*\mathrm{W}}(\mathrm{x_j})) \quad \text{(Eq5.17)}$$

$$\mathrm{I}^- = (\mu_{\mathrm{I}^-\mathrm{W}}(\mathrm{x_j}), \nu_{\mathrm{I}^-\mathrm{W}}(\mathrm{x_j})) \quad \text{(Eq5.18)}$$

where

$$\mu_{I^*W}(x_j) = \left(max\, \mu_{I_iW}(x_j) | j \in C_B\right), \left(min\, \mu_{I_iW}(x_j) | j \in C_C\right)$$

$$\nu_{I^*W}(x_j) = \left(min\, \nu_{I_iW}(x_j) | j \in C_B\right), \left(max\, \nu_{I_iW}(x_j) | j \in C_C\right)$$

$$\mu_{I^-W}(x_j) = \left(min\, \mu_{I_iW}(x_j) | j \in C_B\right), \left(max\, \mu_{I_iW}(x_j) | j \in C_C\right)$$

$$\nu_{I^-W}(x_j) = \left(max\, \nu_{I_iW}(x_j) | j \in C_B\right), \left(min\, \nu_{I_iW}(x_j) | j \in C_C\right)$$

Normalized Euclidean distance is adopted in this paper to measure the separation between alternatives and positive-ideal solution$\mathrm{I}^*$and negative-ideal solutionI^- as $\mathrm{D_i^*}$ and $\mathrm{D_i^-}$:

$$D_i^* = p'_{ij} - I^*$$

$$= \sqrt{\frac{1}{2n} \sum_{j=1}^{n} \begin{bmatrix} (\mu_{I_iW}(x_j) - \mu_{I^*W}(x_j))^2 + (\nu_{I_iW}(x_j) - \nu_{I^*W}(x_j))^2 \\ +(\pi_{I_iW}(x_j) - \pi_{I^*W}(x_j))^2 \end{bmatrix}} \quad \text{(Eq5.19)}$$

and

$$D_i^- = p'_{ij} - I^- =$$

$$\sqrt{\frac{1}{2n}\sum_{j=1}^{n}\begin{bmatrix}(\mu_{I_iW}(x_j) - \mu_{I^-W}(x_j))^2 + (\nu_{I_iW}(x_j) - \nu_{I^-W}(x_j))^2 \\ +(\pi_{I_iW}(x_j) - \pi_{I^-W}(x_j))^2\end{bmatrix}} \quad \text{(Eq5.20)}$$

Step 7 Ranking the alternative instruments based on the relative closeness coefficient.

All instruments are then scored with the relative closeness coefficient RC_i with respect to the positive-ideal solution:

$$RC_i = \frac{D_i^-}{D_i^- + D_i^*} \quad \text{(Eq5.21)}$$

The candidates are then ranked according to the value of RC_i. Higher score indicates more suitability of the corresponding alternative instrument.

The algorithm of Phase 2 is shown in Figure 5.6 and Figure 5.7 and the most suitable instrument is highlighted as the result of instrument selection process.

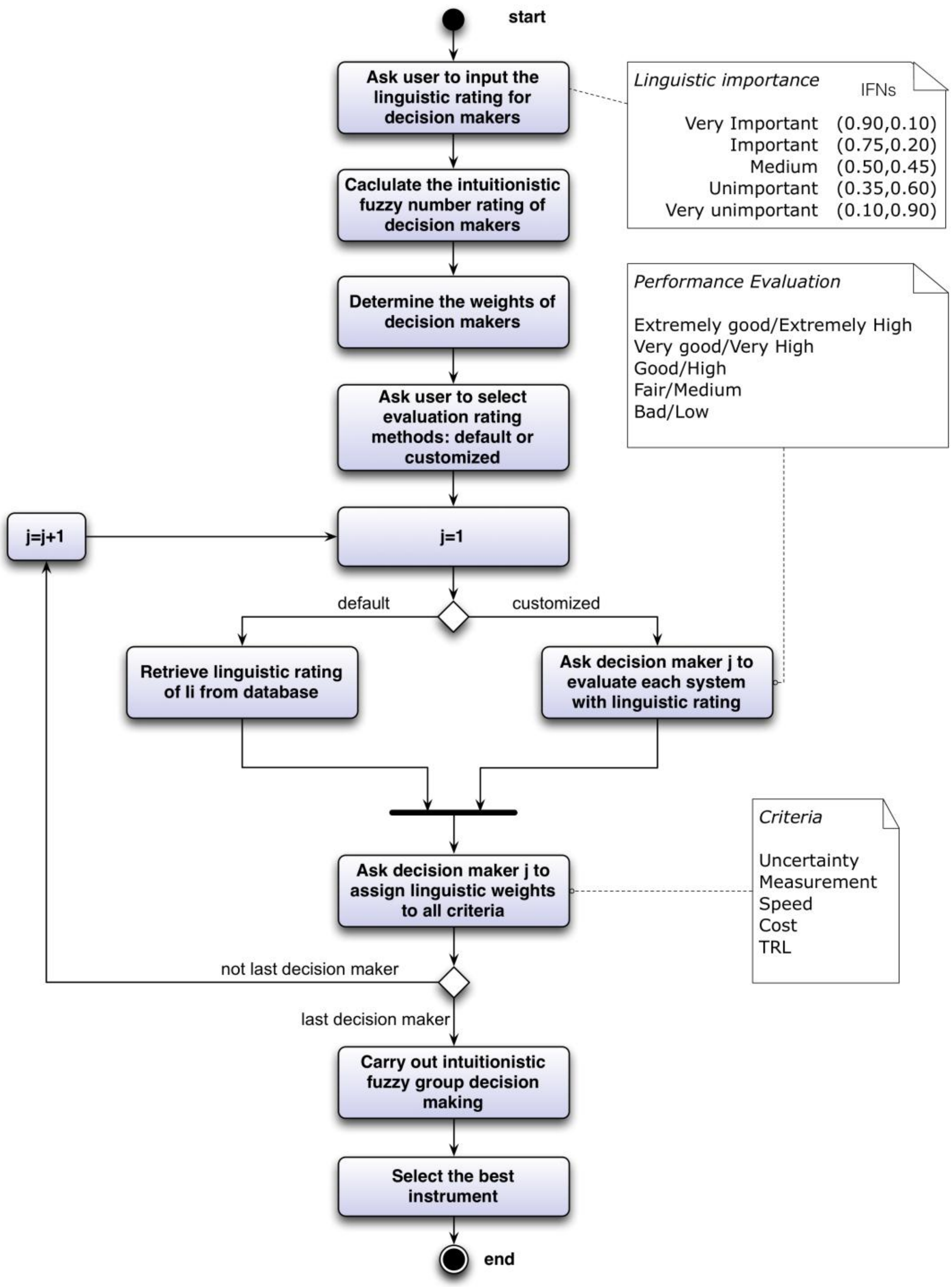


Figure 5.6 UML activity diagram of instrument selection-Phase 2

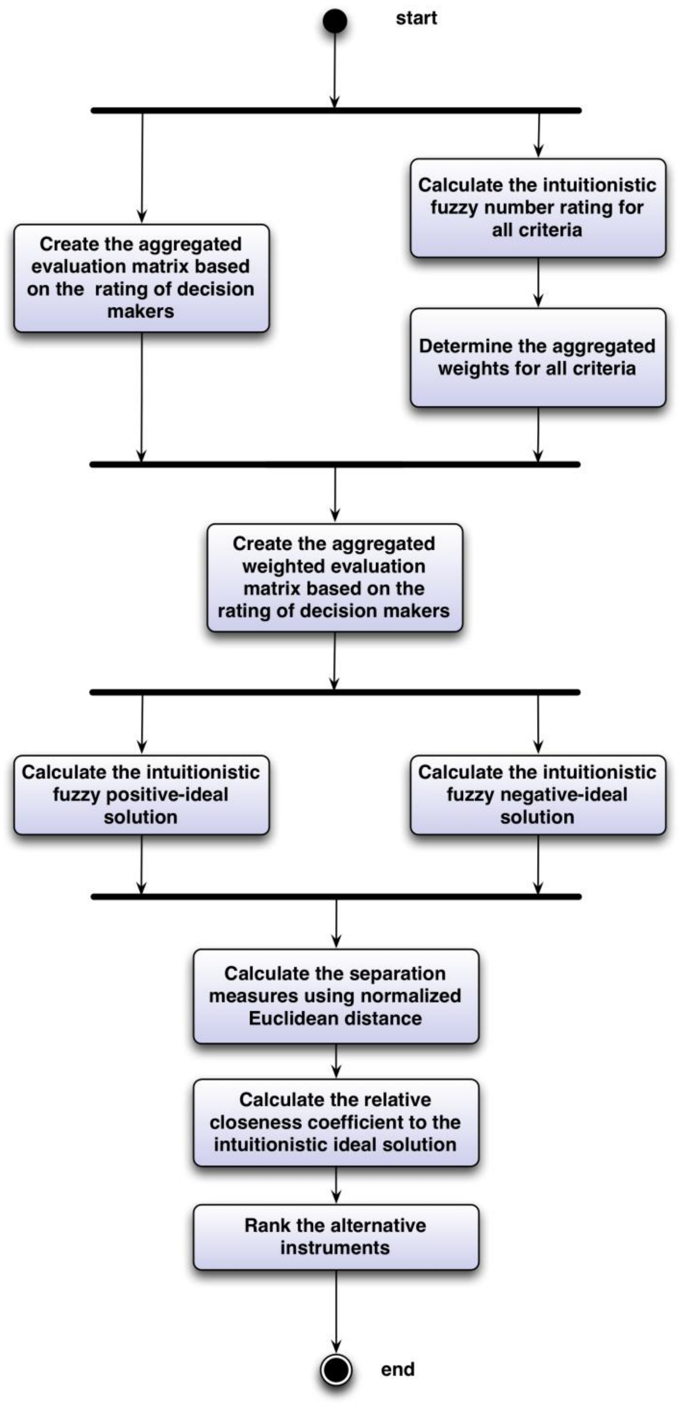


Figure 5.7 UML activity diagram of fuzzy decision making process

5.2.3 Numerical case study

An instrument is required for an inspection task and the crisp MCs is shown in Table 5.2. Available instruments stored in database include 2 laser trackers, 2 laser scanners, laser radar, iGPS and photogrammetry system. The filtration process is implemented as follows:

1) photogrammetry system is removed from CIL due to insufficient range coverage.

2) laser scanners and iGPS are filtered out due to unsatisfied uncertainty requirement.

Under this circumstance, 2 laser trackers and laser radar have remained from Phase 1, as the alternative instruments

I_1: Laser Tracker 1

I_2: Laser Tracker 2

I_3: Laser Radar

One designer (DM_1) and two metrologists (DM_2 and DM_3) are involved in the performance evaluation process based on the four fuzzy MCs：

C_1: Instrument uncertainty

C_2: Overall cost

C_3: Inspection Speed

C_4: TRL

The process of fuzzy instrument selection consists of the following steps:

Step 1 Assigning linguistic importance to DMs and calculating the corresponding weights.

Each DM is assigned with a linguistic importance term shown in Table 5.7 This process is based on the degree of knowledge possessed by DMs regarding specific inspection task and instrument. Corresponding weights are obtained using Eq5.9.

Step 2 Aggregating the decision matrix with respect to the individual performance rating of decision maker.

The performance rating from each DM is shown in Table 5.8. The linguistic ratings are then converted to IFNs using Table 5.5. The intuitionistic fuzzy decision matrix is calculated according to Eq5.10.

Table 5.7 DMs importance and corresponding weights

	DM_1	**DM_2**	**DM_3**
Linguistic importance	Important	Very important	Medium
Weight	0.356	0.406	0.238

Table 5.8 Performance rating of alternatives

Criteria	**Instrument**	**DM_1**	**DM_2**	**DM_3**
C_1 uncertainty	I_1	EG	EG	VG
	I_2	VG	G	VG
	I_3	G	G	F
C_2 cost	I_1	VH	VH	H
	I_2	M	H	M
	I_3	EH	EH	EH
C_3 speed	I_1	VG	F	F
	I_2	G	G	F
	I_3	VG	EG	VG
C_4 TRL	I_1	VG	EG	VG
	I_2	G	VG	G
	I_3	F	G	B

$$
P = \begin{array}{cc}
\begin{array}{cc} C_1 & C_2 \end{array} & \\
\left[\begin{array}{c|c}
(1.000,0.000,1.000) & (0.780,0.118,0.662) \\
(0.764,0.133,0.632) & (0.594,0.302,0.292) \\
(0.661,0.236,0.426) & (1.000,0.000,1.000) \\
\hline
(0.639,0.244,0.395) & (1.000,0.000,1.000) \\
(0.594,0.302,0.292) & (0.746,0.151,0.595) \\
(1.000,0.000,1.000) & (0.553,0.332,0.220)
\end{array}\right] & \begin{array}{c} I_1 \\ I_2 \\ I_3 \\ I_1 \\ I_2 \\ I_3 \end{array} \\
\begin{array}{cc} C_3 & C_4 \end{array} &
\end{array}
$$

Step 3 Calculating the aggregated weight of criteria

The assigned importance by DMs with respect to each criterion is shown in Table 5.9 with converted corresponding IFNs.

The weight matrix is aggregated using Eq5.12 as:

$$
W = [w_{C_1}, w_{C_2}, w_{C_3}, w_{C_4}]
$$

$$
= \begin{bmatrix} (0.861,0.128,0.011) \\ (0.787,0.189,0.023) \\ (0.799,0.170,0.031) \\ (0.576,0.371,0.053) \end{bmatrix}^{T}
$$

Table 5.9 Assigned importance for all criteria

Criteria	DM_1	DM_2	DM_3
C_1	I (0.75,0.2)	VI (0.90,0.10)	VI (0.90,0.10)
C_2	VI (0.90,0.10)	I (0.75,0.2)	M (0.50,0.45)
C_3	I (0.75,0.2)	I (0.75,0.2)	VI (0.90,0.10)
C_4	M (0.50,0.45)	M (0.50,0.45)	I (0.75,0.2)

Step 4 Creating the weighted decision matrix by aggregating matrices *P* and *W*.

With the constructed intuitionistic fuzzy decision matrix *P* and weights matrix *W*, the aggregated weighted decision matrix P' is established using Eq5.11.

$$P' = \begin{array}{cc}
\begin{array}{cc} \quad\quad C_1 \quad\quad\quad\quad & \quad\quad C_2 \quad\quad \end{array} & \\
\left[\begin{array}{c|c}
(0.861,0.128,0.011) & (0.614,0.285,0.102) \\
(0.658,0.224,0.098) & (0.467,0.434,0.099) \\
(0.569,0.334,0.097) & (0.787,0.189,0.024) \\
\hline
(0.511,0.373,0.117) & (0.576,0.371,0.053) \\
(0.474,0.421,0.105) & (0.429,0.466,0.105) \\
(0.799,0.170,0.031) & (0.318,0.580,0.102)
\end{array}\right] & \begin{array}{c} I_1 \\ I_2 \\ I_3 \\ I_1 \\ I_2 \\ I_3 \end{array} \\
\begin{array}{cc} \quad\quad C_3 \quad\quad\quad\quad & \quad\quad C_4 \quad\quad \end{array} &
\end{array}$$

Step 5 Calculating the separation distance of each alternative to positive-ideal solution and negative-ideal solution.

In this case, uncertainty, TRL and speed are considered as beneficial criteria and cost is deemed as cost criterion. Therefore, $C_B = [C_1, C_3, C_4]$ and $C_c = [C_2]$. The intuitionistic fuzzy positive-ideal solution I^* and negative-ideal solution I^- are obtained as:

$$I^* = \{(0.861,0.128,0.011), (0.467,0.434,0.099),$$

$$(0.799,0.170,0.031), (0.576,0.371,0.053)\}$$

$$I^- = \{(0.569,0.334,0.097), (0.787,0.189,0.024),$$

$$(0.474,0.421,0.105), (0.318,0.580,0.102)\}$$

Normalized Euclidean distance is obtained to measure the separation between alternatives and positive-ideal solution I^* and negative-ideal solution I^- using Eq5.19 and Eq5.20 in Table 5.10.

Table 5.10 Separation measure and relative closeness coefficient

Instruments	D^*	D^-	RC
I_1	0.148	0.192	0.565
I_2	0.183	0.162	0.469
I_3	0.228	0.147	0.393

Step 6 Ranking the alternative instruments based on the relative closeness coefficient.

All instruments are then scored with the relative closeness coefficient RC_i with respect to the positive-ideal solution shown in Table 5.10. The ranking order is $I_1 \succ I_2 \succ I_3$and I_1 is selected as the most appropriate instrument in the alternatives since it has the highest RC.

With the purpose of demonstrating the sensitivity of the decision model, a different importance set is assigned to all criteria shown in Table 5.11 and the results are given in Table 5.12.

Table 5.11 Assigned importance for all criteria-case 2

Criteria	DM_1	DM_2	DM_3
C_1	I (0.75,0.2)	I (0.75,0.2)	I (0.75,0.2)
C_2	VI (0.90,0.10)	I (0.75,0.2)	M (0.50,0.45)
C_3	VI (0.90,0.10)	VI (0.90,0.10)	VI (0.90,0.10)
C_4	M (0.50,0.45)	M (0.50,0.45)	I (0.75,0.2)

Table 5.12 Separation measure and relative closeness coefficient-case2

Instruments	D^*	D^-	RC
I_1	0.168	0.165	0.495
I_2	0.182	0.153	0.456
I_3	0.157	0.178	0.531

The inspection speed is considered more important than in the previous case while less importance is given to inspection uncertainty. This shift leads to a clearly different decision as I_3 is ranked first due to its significantly higher rating than I_1 and I_2 in terms of measurement speed. The decision model is successfully aware of this priority change of criteria and reveals the correct selection result.

5.2.4 Summary of instrument selection

In real industrial applications, it is likely that multiple instruments are available and the inspection process can be conducted simultaneously to enhance the efficiency and reduce the time of measurement. The proposed method is capable of coping with this situation by dividing the inspection tasks into groups. Individual selection process can then be carried out resulting in the most suitable instrument for each task. In the case of multiple same instruments, the module intends to employ all available instruments to maximize the use of resource.

5.3 Inspection point selection

After the appropriate instrument is selected in stage 2, inspection strategy is confined by the adopted sampling principle. Many researchers have tried to classify LVM instruments using different criteria (Peggs et al., 2009). As far as data acquisition approach is concerned, they can be categorized as instruments conducting discrete point measurement and instruments employing scanning measurement. For scanning operation, the inspection objective is the actual feature and therefore there is no need to distribute the measurement points. The inspection point selection process is only valid when point measurement is taken place.

It is required in BSI 7172 (1989) that an adequate number of points should be placed on the work piece appropriately to provide a reliable and accurate representation of the geometric feature. Surprisingly, this area has been rarely studied recently in the literature and methods only for CMMs are found (Zhao et al., 2009). The author believes that this was due to the variety of inspection features in real manufacturing environment, which resulted in the strong dependency between the number and location of inspection points and the experience and knowledge of the expert. Lee et al. (1997), Fan and Ming (1998) and Hwang et al. (2002) developed different methodologies to tackle this issue using neuro-fuzzy inference engine and Hammersley algorithm. However, applications of those techniques are limited to certain features and manual operation is required throughout.

Under this circumstance, this module aims to supply general guidance facilitating user to identify the number of measuring points and distribute them appropriately on the workpiece within metrology software.

5.3.1 Number of points

BSI 7172 has laid the instruction for deciding the minimum number of points, shown in Table 5.13. Each feature has a minimum mathematical number of points in order to determine the element. Although increasing the number of sampling points is expected to obtain a static benefit on the result, greater number of points is not consistent with more accurate measurement, which is attributed to the greater noise that occurred in the sampling process. Filtering algorithms can remove the noise to some extent, but this introduces further processes and therefore time.

5.3.2 Distribution of points

Expert experience is desired in this stage despite the presence of general rules for basic features (inspection tasks defined in stage 1). Examples are given in Figure 5.8. Information and guidance for more features can be found in BSI 7172.

Table 5.13 Minimum number of points in BSI 7172

Feature	Minimum number of points	
	Mathematical	Recommended
Line	2	5
Plane	3	9
Circle	3	7
Sphere	4	9
Cylinder	5	12
Cone	6	12

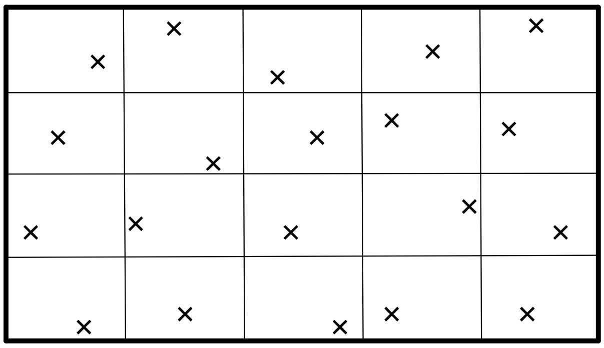
1) Lines: *N* points should be placed into *N* subintervals of the line with random positions to achieve an approximately uniform distribution. In the case that points are too close, *3N-2* subintervals are used and each point is located randomly in the *1st, 4th, 7th* and *(3N-2)nd* subintervals.

2) Planes: The rectangular plane should be divided into $\boldsymbol{N_1} \times \boldsymbol{N_2} (\approx \boldsymbol{N})$ sub-rectangles using regular mesh of lines in order to achieve an approximately uniform 2D distribution of *N* points. Each point is placed randomly within each sub-rectangle and irregular patterns are preferred to identify the periodic distortion of the plane.

3) Circles: *N* points should be placed into *N* equal arcs of the circle with random positions to achieve an approximately uniform distribution. In the case that the circle is distorted with q lobes, N should be chosen greater than q and indivisible by q.

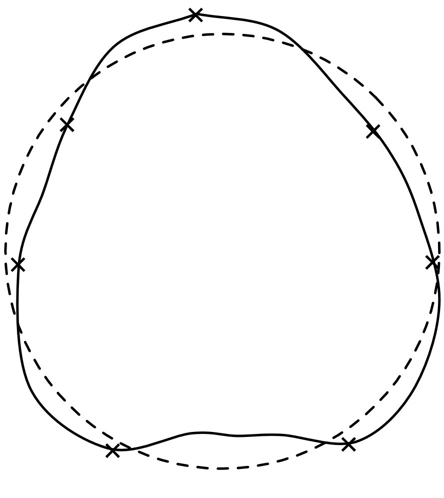
Example. N=5 and the line segment is divided into 13 (3N-2) subintervals. Points are placed randomly at 1st, 4th,7th,9th and 13th subintervals.

(a) A distribution of points on a line



Example. N=20 and the plane is divided into 4x5 (N1=4,N2=5) sub-rectangles. Points are placed randomly in each sub-rectangle.

(b) A distribution of points on a plane



Example. N=7 and q=6. seven uniformly located points detect at least 79% of the lobes.

(c) A distribution of points on a lobed circle

Figure 5.8 Examples of points distribution on line, plane and lobed circle

5.4 Summary

Recent developments of inspection process planning methodology demands instrument selection as a mandatory and vital process, which paves the way for subsequent planning activities. Nevertheless, measurement device selection for large volume metrology is rarely studied with most research efforts focusing on the probe selection for coordinate measuring machines. The large and increasing number of available instruments with a variety of assessing criteria presents a barrier to an applicable selection system.

A two-phased LVM instrument selection system using intuitionistic fuzzy sets combined with TOPSIS method is described in this chapter. Measurability characteristics are first identified with respect to specific inspection task and grouped into quantitative (crisp) and qualitative (fuzzy) attributes. An instrument filtration procedure is implemented in Phase 1 based on the results of assessing crisp MCs. In the second phase, the remaining instruments are ranked using intuitionistic fuzzy group decision-making method. Vague criteria are appropriately assessed in this early stage by taking advantage of linguistic importance and performance rating. A numeric case study shows the process of the proposed approach. Furthermore, the sensitivity of the decision model to the variable priority of the criteria has been successfully demonstrated.

The developed method successfully solved the inherent decision-making problem embedded within the instrument selection process. By means of introducing fuzzy theory into weighting criteria and evaluating candidates, appropriate selection can be achieved at this early stage without requiring uncertain information such as the inspection point distribution and instrument configuration.

Target inspection points are selected and distributed on the part after the sampling strategy of the specific instrument is confirmed. However, only guidance is provided to the user to facilitate the process due to the vast variety of features existing in real manufacturing environment. Expert knowledge and experience is required for an appropriate distribution, which leads to preferable inspection results.

CHAPTER 6 VISIBILITY ANALYSIS FOR LVM APPLICATIONS

6.1 Introduction

Accessibility analysis is deemed as a vital process for CMMs IPP systems, reviewed by (Li et al., 2004, Zhao et al., 2009). This process determines the feasible orientations of the probe for measurement points. Reduced inspection time is achieved when all points are clustered with a minimum number of probe orientations. Moreover, inspection sequence is derived based on the minimization of the changes of part set-up and probe orientation.

This chapter proposes a novel approach focusing on visibility analysis for LVM instruments such as laser trackers and a subsequent clustering process is carried out based on the results of the analysis. With the help of modern CAD software rendering technique, a global visibility cone (GVC) of the target point is calculated without investing enormous computational resource.

6.2 Proposed methodology

As laser tracker is a frameless coordinate measurement system that measures the position of target, the primary objective of the visibility analysis for a laser tracker is to identify the spatial domain in which the base of laser tracker can be placed and the line of sight to the retroreflector located at target positions is maintained while carrying out the inspection. In other words, the analysis is a visual set generation process that finds all view directions in which the given object in the scene is completely visible.

6.2.1 Problem modelling

6.2.1.1 Abstraction of instrument

According to the literature, the straight probe on a CMM was abstracted initially as a half-line emitting from the tip to infinity in nearly all approaches. Other parts such as the ram and stylus were abstracted and considered using different techniques after the initial analysis. It can be reasoned that only if there is enough clearance for the probe and the ram a surface feature is accessible by a straight probe (Spyridi and Requicha, 1990). Otherwise collision will occur between the abstracted half-line and the surface.

As a CMM, however working on optical measuring principle, laser tracker operates by firing a laser beam to the retroreflector located at target position and tracing back the reflected beam to the base. In spite of the different types of retroreflectors, commercial laser tracker system normally use a sphere mounted retroreflector (SMR) with a variety of accessories when the SMR cannot be attached to the target point on inspection part. Serving as reverberator that provides a precisely defined mechanical interface, SMR does not affect the line of sight of the laser beam when its geometrical volume is taken into account and therefore this makes an infinite half-line originating at the target point as a suitable and adequate abstraction of the system. In case that accessory is engaged to measure a point that the SMR cannot be mounted directly, the actual measuring point is relocated with the SMR and the visibility analysis is carried out at the new location as needed.

6.2.1.2 Solution space

Limited by the resolution of the mechanism of the probe head, applied probe on CMM has certain minimum degree while changing the pointing direction. For instance, a common indexable probe is capable of rotating at a resolution of 7.5° about both the horizontal axes A, ranging from 0° to 105°, and the vertical axes B, ranging from 0° to ±180°, which leads to a total of 673 orientations. In addition to that, the final objective of accessibility analysis for CMM is to determine the common orientation/orientations that can measure all inspection points by manoeuvring the column. Although the entire accessibility cone can be calculated using some methodologies (Woo, 1994; Dhaliwal et al., 2003; Liu et al., 2009) the motivation behind this is to select those optimal orientations for inspection. Therefore discretising the accessible space into a set of orientations makes an appropriate approximation and was adopted by most researchers (Lim and Menq, 1994; Limaiem and ElMaraghy, 1997; Ziemian and Medeiros, 1997; Jackman and Parka, 1998; Vafaeesefat and ElMaraghy, 1998; Limaiem and ElMaraghy, 1999; Vafaeesefat and ElMaraghy, 2000a; Vafaeesefat and ElMaraghy, 2000b; Fernández et al., 2008; Alvarez et al., 2008a; Álvarez et al., 2008b).

On the other hand, LVM system such as laser tracker is not constrained by any mechanism in terms of manipulating the direction of laser beam. Rather than searching for common orientations, the ultimate goal of visibility analysis is to identify the 3D configuration space where the base of laser tracker can be located and line of sights to the greatest number of inspection points can be assured. Under this circumstance, approximating the solution space by a finite number of orientations is neither adequate nor precise to represent all possible positions of the instrument unless an enormous number of locations are sampled, whereas it is not computationally realistic. It is then required that continued solution space should be calculated and generated for further clustering analysis and instrument localization.

6.2.1.3 Part discretization

Knowing that most concepts and methodologies presented in the literature are restricted to the analysis of polyhedron drives this research to develop an approach that can be applied regardless of the geometry. In current research, all parts and environmental obstacles are discretized in a set of triangles, known as the Stereolithography (STL) model, to enable the proposed algorithm to work on all types of geometry.

Benefited from small size and easy implementation, STL format has been widely adopted for rapid prototyping and computer-aided manufacturing and supported by many CAD software packages nowadays e.g. CATIA, UG and Solidworks. As a format capable of defining a closed polyhedron, it ignores common CAD attributes such as colour and texture and only presents the part by raw unstructured triangulated surfaces using triangles with the unit normal and vertices, which makes an efficient way of triangulating geometry (Wikipedia). A binary STL file contains 4 byte unsigned integers showing the number of triangular facets and twelve 32-bit-floating point numbers for every facet indicating the normal and 3D coordinate of each vertex (Wikipedia). The facet normal is defined as the normal vector pointing outwards from the solid object, which is valuable information in later analysis. Figure 6.1 shows an example of a typical binary STL file.

```
UINT8[80] – Header
UINT32 – Number of triangles

foreach triangle
REAL32[3] – Normal vector
REAL32[3] – Vertex 1
REAL32[3] – Vertex 2
REAL32[3] – Vertex 3
UINT16 – Attribute byte count
end
```

Figure 6.1 Data structure of a binary STL file (Wikipedia).

As far as the sampling strategy of laser tracker is concerned, measurements are conducted at discrete points on the surface of an object. In other words, only the inspection points are of importance. It is therefore more practical and reasonable to implement visibility analysis on those points respectively instead of considering the entire facet or surface. Individual solution spaces are expected for every inspection point after the analysis process and subsequent stage of finding common spatial domain is taken place based on that information.

6.2.2 Terminology in visibility analysis

In order to make this chapter self-contained, basic concepts and terminologies regarding computational geometry and visibility analysis are given in this section.

Throughout this chapter, a part after triangulated using STL format is a polyhedron denoted by P bounded by a set of connected polyhedral surfaces S, triangle meshes in this case. V_S, E_S, F_S indicate the vertex, edge and face of a triangle S and $\boldsymbol{n}_F$ refers to its outward normal.

6.2.2.1 Review of computation geometry

Convex set: In Euclidean space, an object is convex if for every pair of points within the object, every point on the straight-line segment that joins them is also within the object.

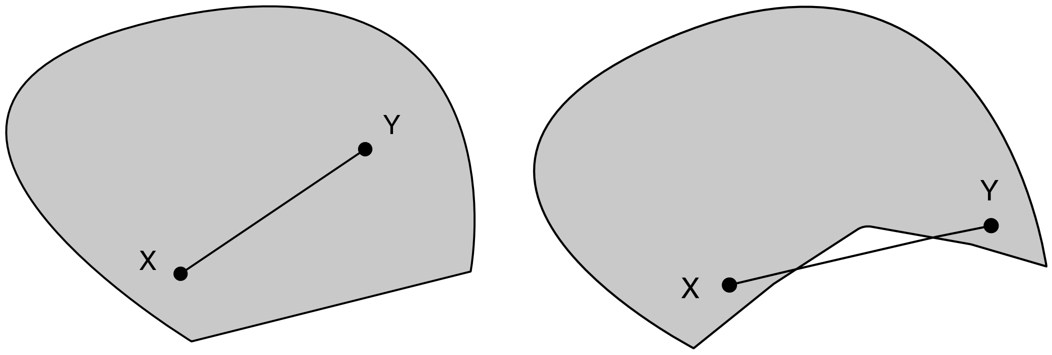


Figure 6.2 An example of convex set

Convex polygon: A polygon is convex if for any two points in the polygon (interior and boundary) the segments connecting the points are entirely within the polygon.

Convex hull: Convex hull is a fundamental and ubiquitous structure in computational geometry. In a vector space, the convex hull for a set of points S_p is the minimal convex set containing S_p. In a two-dimensional case, given a set S_p = $\{p_1, p_2, \ldots, p_N\}$ of points in a plane, the convex hull $H(S)$ is the smallest convex polygon in the plane that contains all of the points of S_p.

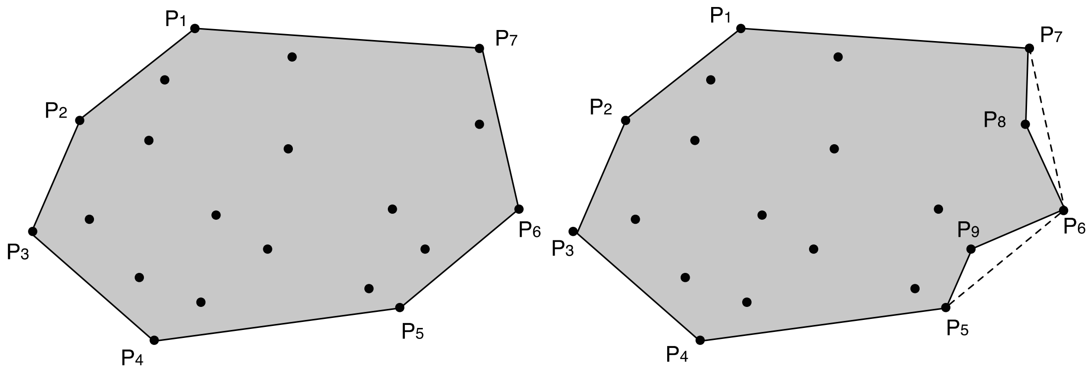


Figure 6.3 Illustration of convex hull

A boundary face F is a hull facet if F is contained within the convex hull of P and on the contrary F is a non-hull facet. Maximum non-hull faces that connect together form a concave region of P. For a 2D example, the polyhedron shown on the right in Figure 6.3 has two concave regions formed by $\overline{P_7P_8}$, $\overline{P_8P_6}$ and $\overline{P_5P_9}$, $\overline{P_9P_6}$. Facets in those two regions are non-hull facets and other facets are hull facets since they are all on the convex hull of the polyhedron.

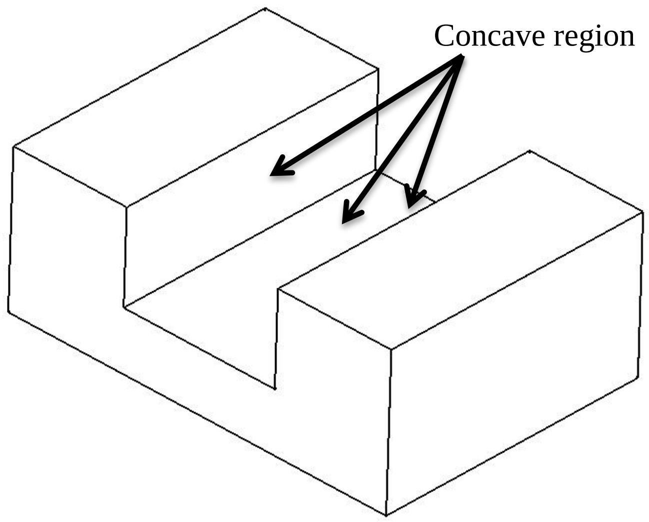


Figure 6.4 An example of concave region

A 3D example is shown in Figure 6.4 indicating a pad with a slot. All facets within the slot are considered non-hull facets and form the slot as a concave region. Facets on other boundary surfaces are all hull-facets since they are on the convex hull of the pad.

Half Space: a facet F on a surface S has a carrying plane that divides the surrounding space into two shown in Figure 6.5: the positive half space of F, namely F^+, is the open space on the side of $\boldsymbol{n}_F$, and the negative half space, namely F^-, is the open space on the other side.

Facing Condition: If a point **p** lies in F^+, facet F is front-facing **p**, and F is back-facing **p** if **p** lies in F^-. In the case that **p** is on the carrying plane of F, F is tangent-facing to **p**. As an illustrative example in Figure 6.5, F is front-facing $\mathbf{p_1}$, tangent-facing $\mathbf{p_2}$, and back-facing $\mathbf{p_3}$.

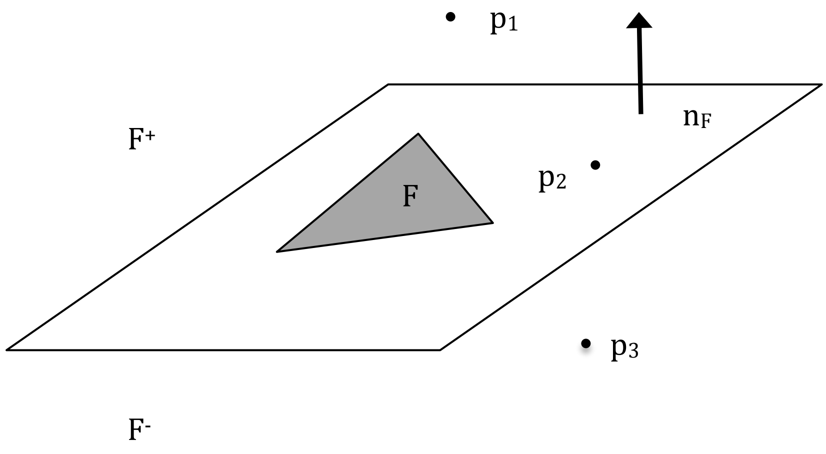


Figure 6.5 Illustration of half space

6.2.2.2 Visibility

For a point **p**, a viewing direction for **p** is modelled as a ray **r** originated at **p**, and the configuration of r is described by a vector **d** with the magnitude equals to R, the range of the instrument.

The followings are definitions for terms used in visibility analysis throughout this research:

a) Visibility Cone (VC)

In Yin et al. (2000) definition, generally, a point **p** on a polyhedron *P* is deemed to be visible in viewing direction **d** if a ray originating from **p** in viewing direction **-d** is free of intersection with *P*. The aggregate of all visible directions is defined as the visibility cone of **p**. It can be defined mathematically as:

$$VC(p, P) = \{d : (p - \lambda d) \cap P = \emptyset, \forall \lambda > 0\} \qquad \text{(Eq6.1)}$$

Figure 6.6 indicates the concept of an exemplary VC of a point **p** on a facet *F*.

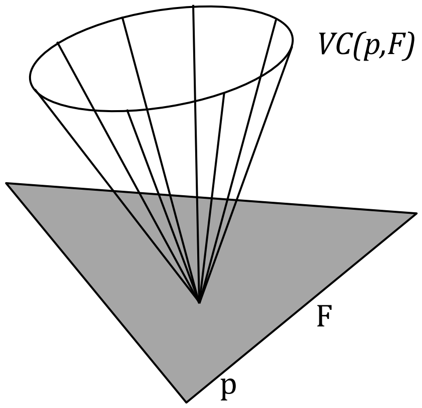


Figure 6.6 An exemplary VC of a point **p** on a facet *F*

b) Local Visibility Cone (LVC)

In order to develop the algorithm used in this research, the local visibility cone of a point **p** on the boundary facet F of a polyhedron P is denoted by $LVC(p, F)$, which indicates all the directions that are not self-occluded by facet *F*. It can be defined mathematically as (Yin et al., 2000):

$$LVC(p, F) = \{d: (p - \lambda d) \cap F = \emptyset, \forall \lambda > 0\} \quad \text{(Eq6.2)}$$

If the outward normal of the facet F is **n,** the LVC of the point **p** can also be defined as:

$$LVC(p, F) = \{d: \angle((p - \lambda d), n) \leq \pi/2, \forall \lambda > 0\} \quad \text{(Eq6.3)}$$

and also can be written as:

$$LVC(p, F) = \{d: (p - \lambda d) \cdot n > 0, \forall \lambda > 0\} \quad \text{(Eq6.4)}$$

One can find that this definition shows that the $LVC(p, F)$ is equivalent to the half space F^+, defined by the carrying plane of facet *F*.

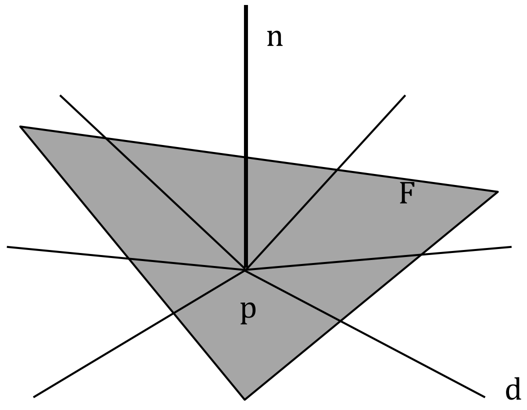


Figure 6.7 An exemplary LVC

c) Global Visibility Cone (GVC)

Local visibility cannot assure the point can be viewed while considering the entire part and therefore a more constrained visibility cone is needed. The global visibility cone of a point **p** on the boundary facet F of a polyhedron P is denoted by $GVC(p,P)$, which indicates all the directions that are not occluded by both F and all other facets on *P*. It can be defined mathematically as (Yin et al., 2000):

$$GVC(p,P) = \{d \in LVC(p,F) : (p + \lambda d) \cap P = \emptyset, \forall \lambda > 0\} \quad \text{(Eq6.5)}$$

and it is clear that

$$GVC(p,P) \subseteq LVC(p,F) \quad \text{(Eq6.6)}$$

However, not only is the polyhedron that has the target point considered in this work, but also the entire working environment is taken into account for the visibility analysis. All objects in the working environment can then be denoted by a set of polyhedrons P_S since they are formatted using STL model. The comprehensive GVC is hereby defined as:

$$GVC(p,P_S) = \{d \in LVC(p,F) : (p + \lambda d) \cap P_S = \emptyset, \forall \lambda > 0\} \quad \text{(Eq6.7)}$$

d) Measurable Space (MS)

MS(p,I) is the measurable space of instrument I located at point **p**, which is the region bounded by the sphere S^2 with radius *R*, . Having located instrument *I* at point

p, any point within the measurable space (volumetric coverage) of *I* is deemed as measurable and inversely, **p** can be measured by *I* located at any location within *MS(p,I).* Figure 6.8 shows an illustrative example revealing the measurable space of a laser tracker. For better illustrating purpose, the scale of the measurable space is modified to achieve comparable results with respect to the dimension of the instrument.

Figure 6.8 Measurable space of a laser tracker

e) Instrument Configuration Space (ICS)

The $ICS(p,P_S)$ is defined as the spatial region, where a point **p** on a polyhedron of the set P_S is visible from any point within this space using a specific inspection instrument. It can be derived by calculating the intersection of $VC(p,P_S)$ and *MS(p,I)* and mathematically defined as:

$$ICS(p, P_S) = VC(p, P_S) \cap MS(p, I) \qquad \text{(Eq6.8)}$$

where *MS(p,I)* is the measurable space of instrument *I* located at point **p**. Figure 6.9 gives an 2D example of the relation between *VC* and *MS* of a point **p** located on facet *F*. The *MS* is shown by the blue sphere originated at **p** with the radius of the range of a specific instrument. The visible space of **p** is defined by the positive half space F^+, namely the *VC*. The *ICS* is then indicated by the green intersection region of *MS* and *VC*. In order to measure the target **p** in this particular case, the chosen instrument is required to be located within the green space to guarantee the visibility.

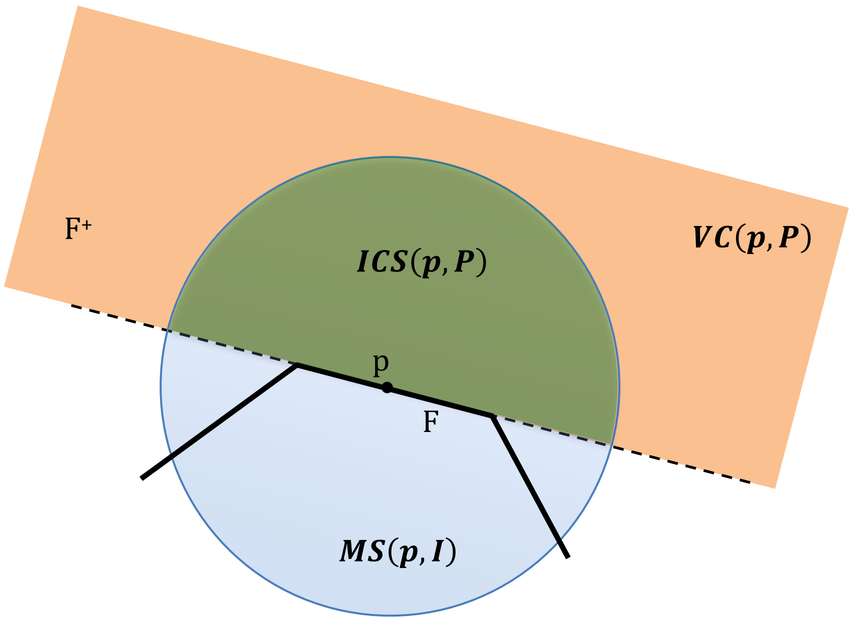


Figure 6.9 A 2D example of instrument configuration space

f) Configuration Space Obstacle (CSO)

The above configuration space can be easily defined when there is no obstacle blocking the line of sight of the instrument. Nevertheless in real application the *ICS* is confined due to the existence of different objects. *CSO* is a well-applied definition in robot planning presenting the set of configurations at which the robot intersects all workspace obstacles (Lozano-Perez, 1983). Yin et al. (2000) adopted this definition and developed the approach computing *LVC* and *GVC* based on the intersection of *CSO* and configuration space (C-space). In their method, every obstacle facet F_{Oi} of **p** maps to a region of S^2, which is the unit sphere at point **p**, defined as:

$$CSO(p, F_{Oi}) = \{d \in S^2: (p + \lambda d) \cap F_{Oi} \neq \emptyset, \forall \lambda > 0\} \qquad \text{(Eq6.9)}$$

In this research, the unit sphere is replaced by the measurable space of a specific instrument at point **p**, defined as:

$$CSO(p, F_{Oi}) = \{d \in MS(p, I): (p + \lambda d) \cap F_{Oi} \neq \emptyset, \forall \lambda > 0\} \quad \text{(Eq6.10)}$$

The advantage of this substitution is that the subsequent computation of visibility cone is simplified significantly. On the one hand, this new strategy eliminates the procedure of projecting the mapped region on S^2 back to the measurable space obtaining the final configurable space of the instrument. On the other hand, obstacles outside the measurable space are not considered while carrying out the mapping process, which consequently saves enormous computational resource.

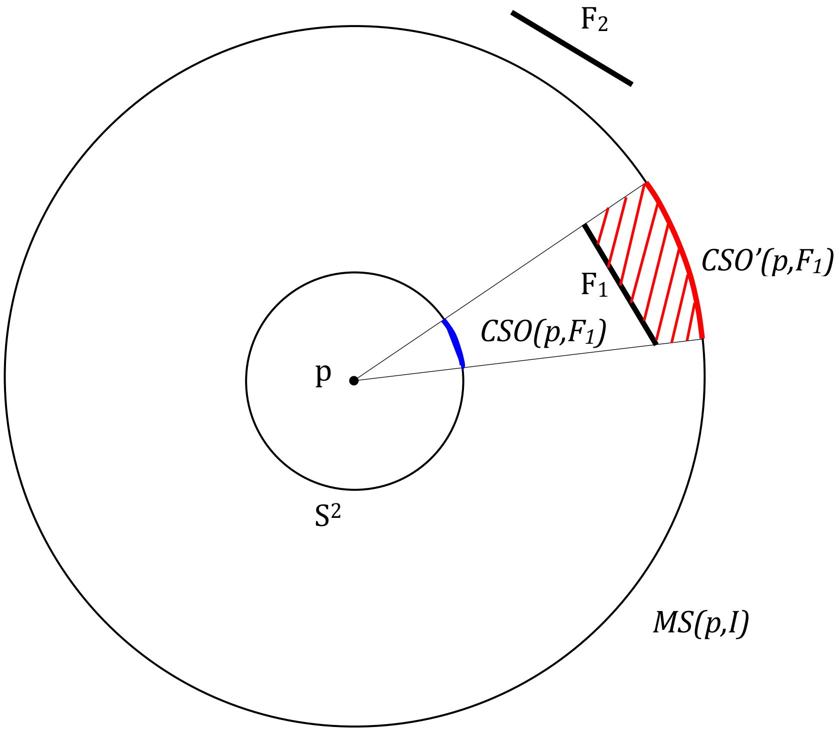


Figure 6.10 2D example of configuration space obstacle

Figure 6.10 shows a 2D illustrative example of the *CSO* using Yin's method and the approach proposed in this research respectively. Following Yin's algorithm, obstacle F_1 is first mapped on the unit sphere S^2 constructing the $CSO(p,F_1)$ coloured in blue. Sphere projecting algorithm is applied in the mapping process and used again to project the $CSO(p,F_1)$ to the space that the instrument is able to cover. Obstacle F_2 is analysed in the same way with the obtained $CSO(p,F_2)$. Instead of using the unit sphere, $MS(p,I)$ is used by this novel approach to avoid the repeated mapping process. Since $MS(p,I)$ represents the volumetric coverage of the instrument, projecting the obstacles directly onto $MS(p,I)$ can efficiently construct

CSO'(p,F$_1$*)* shown by the red arc. The space where line of sight of the instrument is blocked by obstacle F_1 is then obtained shown by the red shadow section. That is to say the instrument can be located anywhere in *MS(p,I)* apart from this shadowed section. Despite F_2 is another potential obstacle it is not considered in the analysis process as the location of F_2 is out of the measurable space coverage of the instrument. The approach benefits this from saving the large amount of computational power required while real working environment is taken into account since all obstacles outside the instrument coverage are eliminated from the expensive sphere projecting process at this early stage.

g) Local Instrument Configuration Space (LICS)

Denoted by *LICS(p,F)*, it defines the spatial region where the point **p** on face *F* can be measured without occlusion by *F* itself using specific instrument e.g. a hemisphere originated at **p** with radius *R:*

$$LICS(p,F) = \{d \in MS(p,I): d \cdot n \ \geq 0\} \qquad \text{(Eq6.11)}$$

where *n* is the outward normal of face *F* at point *p*.

h) Global Instrument Configuration Space (GICS)

Denoted by *GICS(p,* P_S*)*, it defines the spatial region where the point **p** on face *F* can be measured without occlusion by both *F* itself and all other faces in the working environment using specific instrument:

$$GICS(p,P_S) = \{d \in LICS(p,F) : (p + \lambda d) \cap P_S = \emptyset, \forall \lambda > 0\} \qquad \text{(Eq6.12)}$$

and it is obvious that

$$GICS(p,P_S) \subseteq LICS(p,S) \qquad \text{(Eq6.13)}$$

i) Occluded Face Set (OFS)

Since computing individual *CSO* of each face is straightforward using spherical projecting technique, the major complexity and complication in the computation process is brought by the vast number of faces existing not only on the polyhedron itself but also surrounding objects. Time and memory required to conduct the computation will increase significantly as all objects are taken into account. Under this circumstance, it is desired to remove the faces not occluding the target *p* as early as possible to save computational resource in intersecting possible configuration space to obtain configuration space obstacle. Criteria of deciding a face is obstacle or not should be established as well as corresponding algorithm to accelerate and optimize the process.

6.2.3 Assistant techniques for visibility analysis

6.2.3.1 Visibility culling techniques

Visibility culling techniques were employed to decrease the number of faces successfully by Yin et al. (2000; 2002) and Alvarez et al. (2008a; 2008b) such as view-frustum culling, back-face culling, and occlusion culling. The aims of all visibility culling techniques are consistence, which is to identify the faces non-visible to the target and then remove them from the obstacle set, eventually not being computed. The most obvious reason that a face is invisible to the target is because the location of the face is beyond the boundary of the viewable field. Yin et al. (2000) used view-frustum culling to tackle those faces outside the field of view. There are two scenarios while applying the view-frustum culling: (1) if the target p is in the interior of a face *F*, the positive half space, F^+, is the field of view from *p*; (2) if *p* is on the boundary of *F*, the field of view from *p* is the aggregated region of the positive half spaces of all neighbouring surfaces.

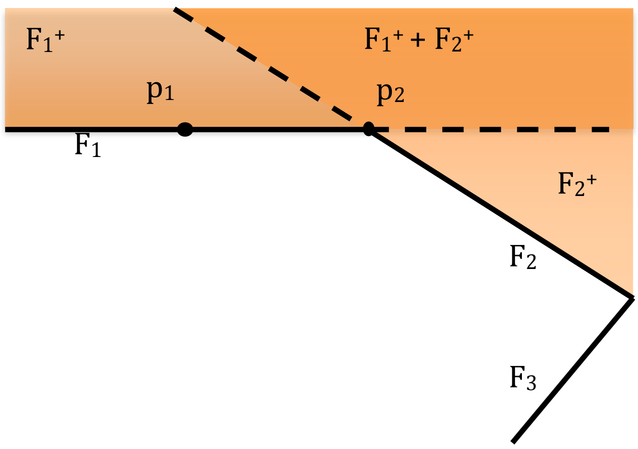


Figure 6.11 View-frustum culling technique

Figure 6.11 gives the basic explanation of the view-frustum culling technique. As far as p_1 is concerned, the field of view is the positive half space F_1^+ since it is in the interior of F_1. F_2 and F_3 will not be considered as occluded faces as they are non-visible to p_1. p_2, however, is located at the edge of F_1 and the field of view is the aggregated space of F_1^+ and F_2^+. Therefore F_3 will be eliminated from the occluded face set, as it is not in the viewable region of p_2.

Since only closed objects are considered with target points on the boundary, back-face culling technique eliminates all faces back-facing the target. Back-facing condition of a surface is guaranteed if the angle between the vector from any point on the surface to the target point and the surface normal is greater than $\pi/2$. Additionally, a face is non-visible to the target if there is any other face, namely occluded face, existing between them. Occlusion culling was adopted by Yin et al. (2000) to detect and remove those faces with distance-based criterion and temporal coherence. Having projected objective faces to a tangent plane of the unit sphere that is parallel to the normal vector of the target face, overlap tests and depth tests were conducted as 2D image-space operations to estimate high-level occlusion using hierarchical occlusion maps. Depending on the different applications, the field of view was determined by the applied effectors e.g. probes and machining tool heads.

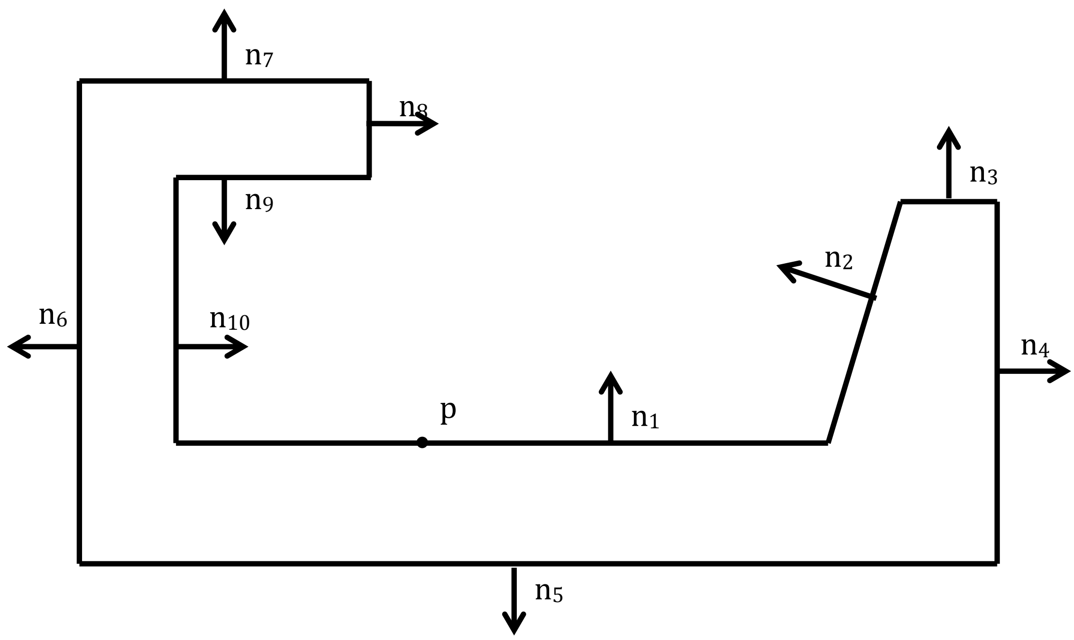


Figure 6.12 Visibility analysis employing different visibility culling techniques

The example shown in Figure 6.12 explains and demonstrates the difference of those visibility culling techniques. There are ten faces in this case with normal vectors shown by the arrows and the target point p is located at F_1. It is required to create the obstacle set in order to carry out the following computation of visibility cones. All faces are initially included in the set and then visibility culling techniques are conducted. F_5 is removed by view-frustum culling since it is in the half negative space of F_1. F_3, F_4, F_5, F_6, and F_7 are eligible to the back-facing conditions and thereby removed from the set. F_3, F_4, F_5, F_6, and F_7 can also be considered as occluded faces due to the fact that any line from p to the surface can reach the surface without penetrating other surfaces or itself. Note that in this particular case, different visibility culling methods are capable of identifying same faces that should be removed. Depending on the complexity of conducting relative techniques, view-frustum culling is carried out first followed by back-face culling and then occlusion culling. As a result, removing target faces by simpler methods can assure that minimum computation resource is spent.

6.2.3.2 Occluded faces on polyhedron boundaries

Liu et al. (2009) summarized the occluded faces need to be considered on polyhedron boundaries from the computation geometry's point of view. Gaussian map and Visibility map were used to represent the visibility of an entire face. For a single polygonal face, the LVM is defined by the closed hemisphere on the unit sphere that has the pole of the normal vector of the face. That is to say only faces in the positive half space are taken into account after the construction of LVM, which has the effect equivalent to view-frustum culling. Having eliminated self-occlusion, GVM is created thereafter to identify the set of spherical points on the unit sphere that indicates view directions free of global occlusion. GOM was then accordingly established considering two categories of faces: potentially faced facets and connected facets.

Property 1. The rays starting from a hull facet F of the polyhedron P can always go to infinity. The rays emanating from a non-hull facet F of P will either go to infinity or be blocked first by another non-hull facet that 'potentially faces' F (Dhaliwal et al., 2003; Liu and Ramani, 2007).

This proved property indicates the fact that if a face F is on the convex hull of the polyhedron the visibility will not be affected by other faces rather than itself. By contrast, if F is in a concave region of the polyhedron, the visibility is confined by faces in the same concave region. It also implies that faces on the convex hull of the object are not taken into account as potentially occluder of *F*. Figure 6.13 shows a pocket with hull facets coloured in yellow and non-hull facets in red. As far as a hull facet in this example is concerned, no other facets affect its viewable space except itself. For a non-hull facet in the red region, the visibility is only compromised by other facets in the same concave region.

A pair of faces is deemed as potentially facing each other and included in the *OFS* when the following conditions are satisfied: (1) the faces are involved in the same concave region; (2) the faces are not coplanar; (3) the faces are both presented fully or partially in each other's positive space. Detailed proof can be found in Liu et al. (2009).

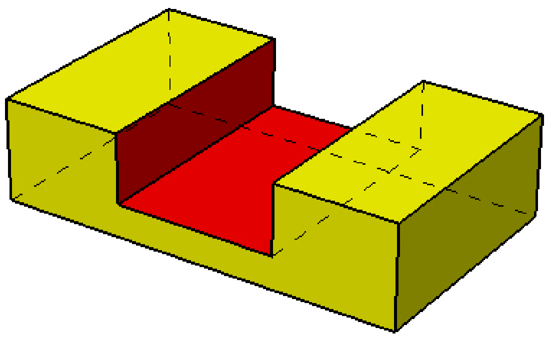


Figure 6.13 Hull facets and non-hull facets on a pocket

Property 2. If a face *F'* shares a convex edge with $F \in S$, no set of parallel rays emanating from *S* can reach *F'* without self-occlusion (Liu et al., 2009).

This property allows the removal of a convex-connected surface from the *OFS*. For faces sharing a concave edge, they are considered as neighbouring occluder for each other and should be maintained in the *OFS*.

A 2D illustrative example is given in Figure 6.14. The boundary of the given polyhedron is consisted of lines from $l_1$to l_9 and the target p is located on l_1. Finding the *OFS(p)* is converted to identifying the *OFS*(l_1). According to the pre-defined property, l_3, l_4, l_5 and l_6 are on the convex hull therefore they are not potentially facing l_1. l_2 is not considered as occluded neighbouring face since it shares a convex edge with l_1, resulting *OFS(p)*={ l_8, l_9}. Some 3D examples have been analysed successfully by Liu et al. (2009) and are shown in Figure 6.15 (a)(b)(c)(d)(e)(f).

In each example, target surfaces are indicated in red while neighbouring occluders are shown in yellow and potential occluders in green. It is clear that in all six models most faces on the polyhedrons are not considered as occluders, which reduces further computation enormously. The analysis process was carried out initially by calculating the convex hull of each polyhedron with both hull facets and non-hull facets tagged. Neighbouring occluders and potential occluders were identified thereafter according to the criteria mentioned above. Figure 6.15(a) and (b) are the cases that only have target faces. There is no neighbouring occluder since all connect faces shares convex edges with target faces and in addition potential occluder does not exist due to the fact that target faces are all on the convex hull of the polyhedron. In (c), (d), (e) and (f), on the contrary, all target faces are in the concave regions of selected models resulting in the existence of both neighbouring occluders and potential occluders.

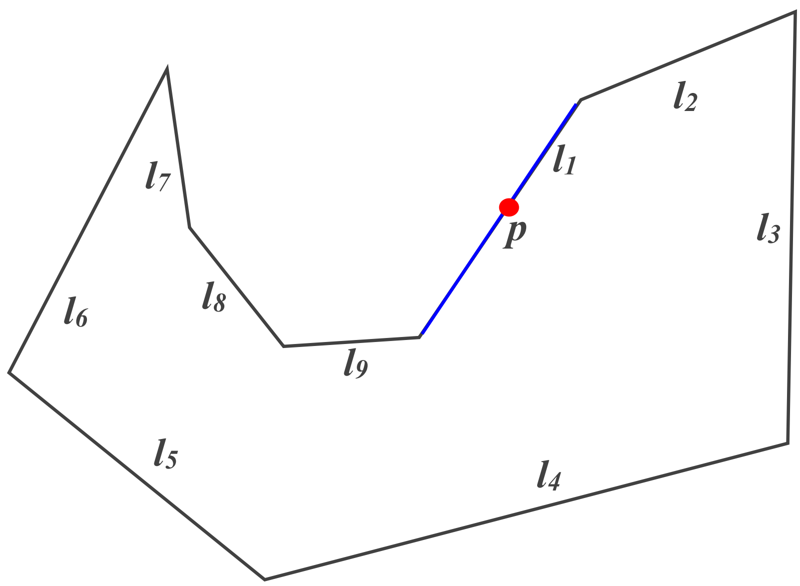


Figure 6.14 Example of identifying occluded faces

The identification of convex hull area of a polyhedron is beyond the scope of this research and further information can be found in the literature Liu et al. (2009).

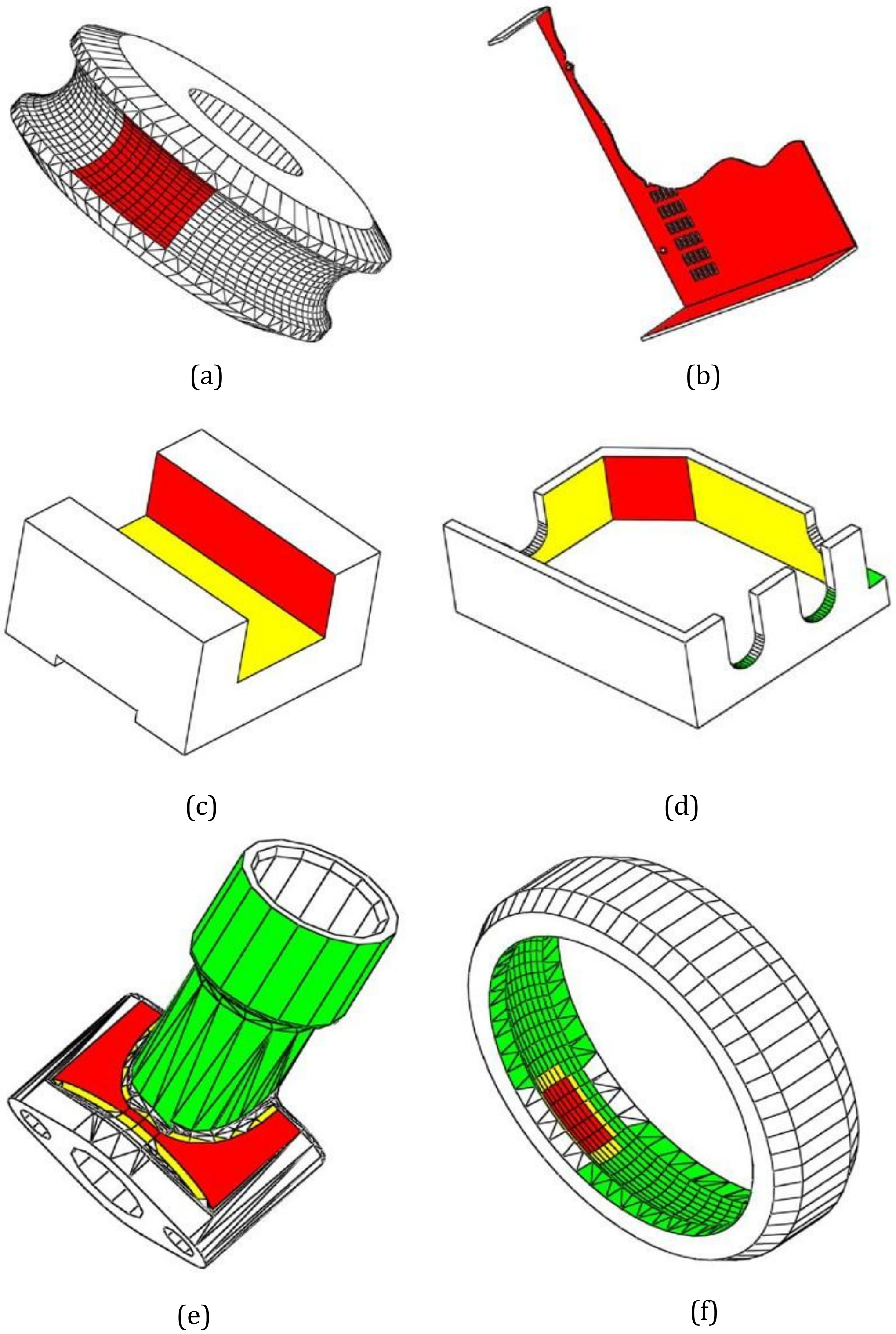


Figure 6.15 Examples of neighbouring occluders and potential occluders Liu et al. (2009)

6.2.3.3 Bounding volume technique

Although the method proposed by Liu et al. (2009) is less complicate and more efficient than the visibility culling techniques adopted by Yin et al. (2004), it only can cope with obstacles on the same polyhedron where the reasoning of the properties were based on. Nevertheless, one unique advantage of this proposed approach is the consideration of entire working environment while analysing the visibility of the selected point. The number of surrounding objects may be vast in an inspection workshop, including fixtures, gigs, other products and instruments that may compromise the visibility of the target. Under this circumstance, for surfaces on surrounding obstacles, an effective way is needed to carry out the identification of occluders. Depending on the required accuracy level, different types of bounding volume are applicable to reduce the complexity of the analysis. Applicable visibility culling techniques are engaged afterwards to tackle those simplified representatives, resulting in the global occluded faces set.

Bounding volume interference detection has been successively applied in computer graphics, computational geometry, detecting tool interference in NC machining and testing collision of CMM probe with product (Klosowski et al., 1998; Ding et al., 2004; Ilushin et al., 2005; Alvarez et al., 2008a). It is a method that generates a simple closed volume, normally simple geometrical elements such as cubes and spheres to enclose a complex object or a set of objects. Computational operations can then be carried out on those simple geometries with enhanced efficiency and speed since checking interface or overlap of all object primitives against each other is avoided. It has been proved that interface or overlap between objects does not exist if the bounding volume does not interface or overlap (Alvarez et al., 2008a). Having adopted this theory into this research, one can claim that the object is not blocking the viewing direction as long as the bounding volume of it is not penetrated by the imagined ray, which emits from the target point. Furthermore, an object can be simplified by its bounding volume while being analysed for potential occluders, which results in only the surfaces of the bounding volume are considered rather than the real geometry elements of the object. In spite of the decreased significant amount of computation, the bounding volume should be carefully selected since it actually enlarges the object therefore reducing the viewable space.

There are a variety of bounding volumes such as sphere, ellipsoid, cylinder, capsule, box, discrete oriented polytope (DOP) and convex hull. The first five volumes are just basically the corresponding geometries containing the object while DOP and convex hull are more complicate groups of geometries but represent the object more accurately. Since all objects in this research have been converted into polyhedrons, bounding volumes including sphere, ellipsoid, cylinder, and capsule are triangulated in order to maintain the consistence in terms of entity type. Figure 6.16 gives the general ideal of bounding volume using three element geometries respectively, namely sphere, cylinder, and box, to include a set of points.

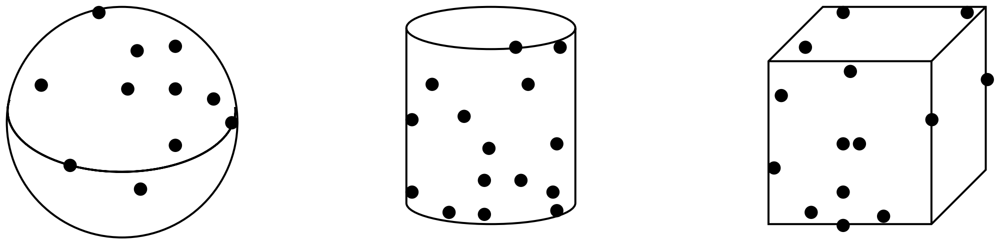


Figure 6.16 Bounding volume examples

The DOP is defined as a generalization of axis aligned bounding boxes (AABB) containing the object, constructed by taking a number of hyperplanes in discrete directions (Wikipedia). As shown in Figure 6.17, the object is presented more precisely as the number of hyperplanes increases, which leads to longer computational time and more required memory.

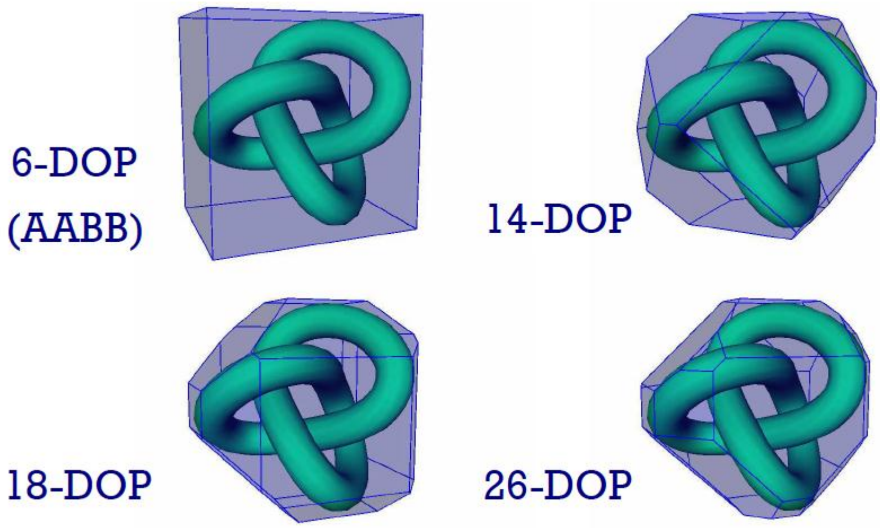


Figure 6.17 Different DOPs (Kimmerl, 2005)

One bounding volume is able to contain a number of objects and vice versa, one object can be represented by several bounding volumes to simulate the object more accurately. Therefore simplifying the object by a number of bounding boxes can achieve comparable accuracy with DOP but requires less computational resource.

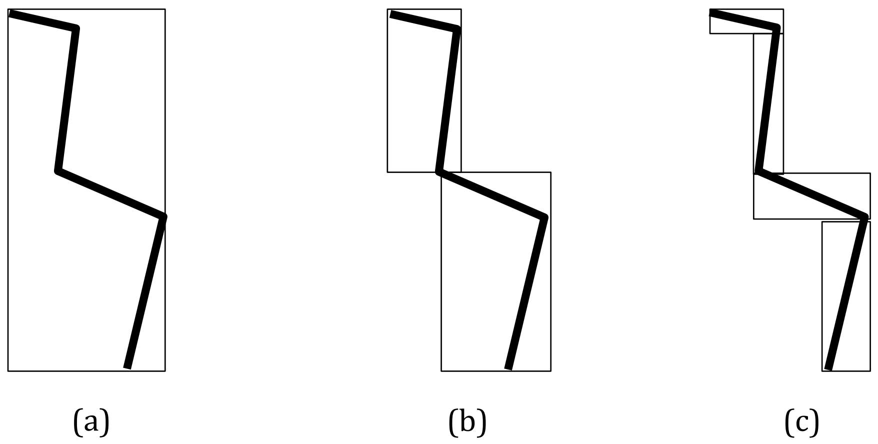


Figure 6.18 More details are obtained as the number of bounding boxes increases

A 2D example is shown in Figure 6.18 indicating the benefits of more bounding boxes representing the same object. More details regarding the geometry of the object are given in (b) and (c) than (a). The faces of employed bounding boxes are inputted to the occlusion analysis instead of the objects. Although four boxes are used in this example, only 24 (4 × 6) rectangular faces are taken into account regardless of the complexity of the object, which remarkably economizes the computational time and cost.

The different number of bounding boxes reflects the altered accuracy level as well as the result of occlusion analysis. Figure 6.19 reveals the comparison of occluded regions caused by different bounding volumes. For a target *p*, *l* is the occluded region when four bounding boxes are used and ($l+ \Delta l$) is the occluded region when only one bounding box is employed. It is anticipated that the latter bounding strategy is less accurate than the former, which has the exact same occlusion result as the object in this instance.

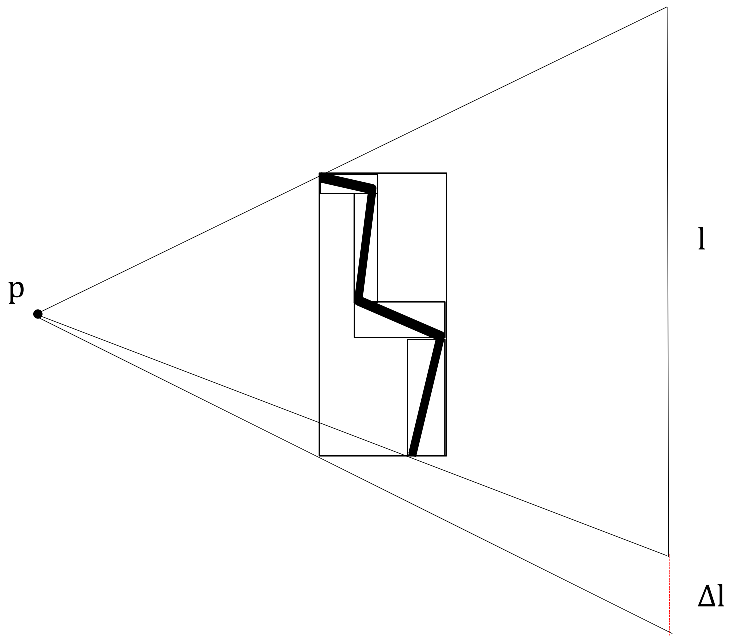


Figure 6.19 Compromised viewable space caused by different bounding volumes

In this research, the decision of bounding volume strategy is strictly determined by the user according to the accuracy of planning and the actual working environment. It is preferred that not having obstacles too close to the line of sight to avoid potential interruption. Therefore in most cases surrounding objects are not necessarily presented in details and it is adequate to simplify those geometries by a number of bounding boxes. An example is shown in Figure 6.20 using three bounding boxes to represent the real geometry of a laser tracker. Only 18 faces are then analysed with acceptable loss of accuracy rather than the complicated geometry of the laser tracker.

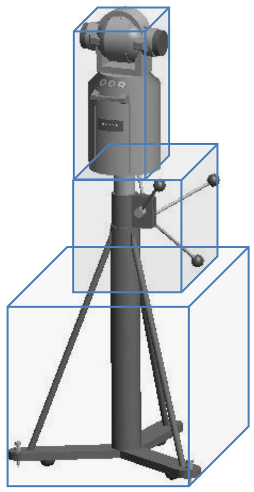


Figure 6.20 A laser tracker simplified by three bounding boxes

The most accurate method, however the most computational expensive one, is to construct the 3D convex hull of the object and then analyse the occlusion of all hull faces. Most popular computation methods for constructing convex hull of a 3D object calculate the minimum convex set containing all points that constitute the object whereas all the vertices of the meshes are considered for a polyhedron object. Static algorithm, incremental construction algorithm, and using a triangulation to get a fully dynamic computation are the three approaches supplied by the open source software Computational Geometry Algorithms Library (CGAL). Details regarding those algorithms are beyond the scope of this research and can be found at the online manual of CGAL (CGAL). Figure 6.21 shows the computed convex hull of a 3D model made of 192135 points. The spent time of this computation was 0.18s by the static approach, 1.90s and 6.80s by the dynamic and incremental approaches respectively with a computer that had 64bit Intel Xeon 2.27GHz processor and 12GB of RAM (Hert and Schirra).

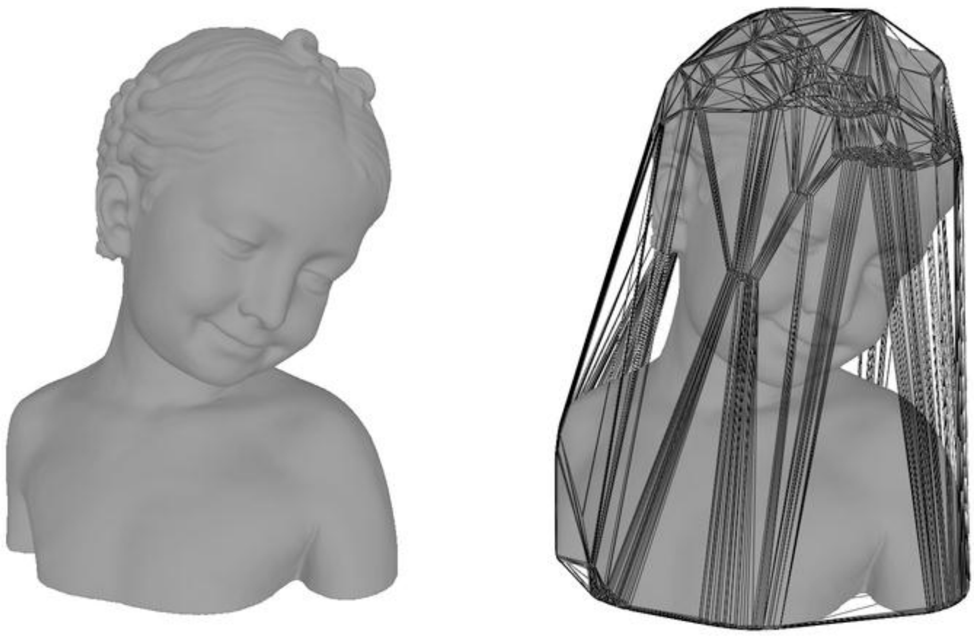


Figure 6.21 The constructed convex hull of a model that consists of 192135 points (Hert and Schirra)

6.2.4 Algorithm for the computation of instrument configuration space

Having all the terminology and definitions explained in the previous section, the following is the algorithm for computing the *ICS* of a point *p* on a polyhedron *P* in a set P_S, which comprises all objects in the scene. STL format has been applied to all faces enabling the algorithm suitable to all types of geometries.

Step1. Generate *MS(p,I)* with respect to engaged instrument.

Once the instrument is selected in stage 2, *MS(p,I)* is constructed with instrument range *R* at the point **p**.

Step2. Construct *OFS(p)* within working environment.

Let $O= \{O_1,O_2,...O_n\}$ denote the set that includes connected or unconnected polyhedron objects within *MS(p,I)*, while every object O_i is bounded by a set of connected faces, $O_i= \{F_{i1}, F_{i2,...,}F_{im}\}$. *OFS(p)* is then identified by assessing O_i with respect to the criteria described previously. As O_1 denotes the object that contains the target point, the calculation of convex hull of O_1 is initially carried out categorizing hull facets and non-hull facets into two groups. Neighbouring occluder and potential occluder are identified afterwards if **p** is located on a non-hull facet. Bounding volume is selected and applied by the user to $O_{2,}$ $O_{3,}$ ... O_n, resulting in new facets set of all chosen bounding volumes $O_i'= \{F_{i1}', F_{i2}',...,F_{im}'\}$. Visibility culling is taken place on the new generated facets and filtered bounding volume facets are subsequently included in *OFS(p)*.

Step 3. Compute *CSO(p,O)* based on *OFS(p)* within *MS(p,I)*.

CSO(p,O) is the aggregated space of all individual *CSO(p,F)* with respect to all facets in *OFS(p)*.

$$CSO(p, O_i) = \cup_{F_{ij} \in O_i} CSO(p, F) \qquad \text{(Eq6.14)}$$

Yin et al. limited *CSO(p,F)* as a spherical image on the unit sphere and Liu et al. used similar constraint to reduce the complexity of the computation. It was reasonable to perform the union operation of individual *CSOs* on the unit sphere in their application since the scale of the space was limited. Space between the target point and the obstacle was considered as non-viewable based on their methods. In this research this space is however, instrument configurable due to the large scale of the working volume and the consideration of all objects in the environment such as gigs and fixtures. Although line of sight in certain space behind the obstacles is blocked, the instruments may still be located between the obstacles and the target point.

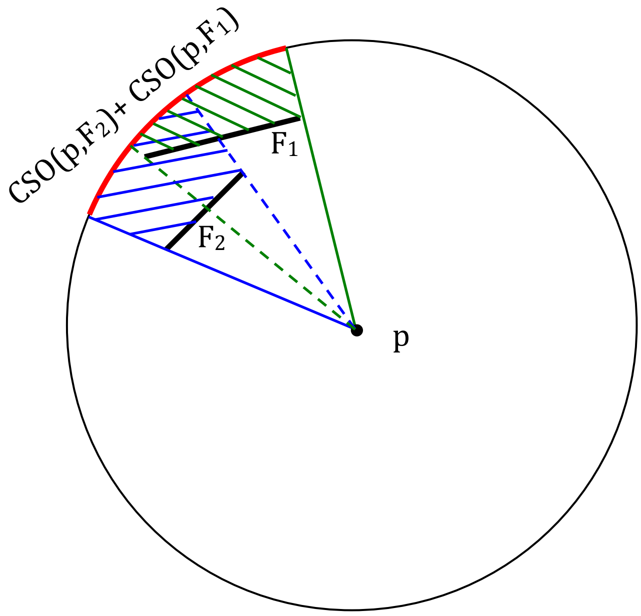


Figure 6.22 Comparison of non-viewable space due to obstacles

Figure 6.22 indicates the difference between the *CSO(p,O)* derived by the union of spherical image and the union of 3D space. If only spherical image is computed, the red arc denotes the union of $CSO(p,F_1)$ and $CSO(p,F_2)$, resulting in the entire space bounded by $CSO(p,F_1)+CSO(p,F_2)$ to **p** is not instrument configurable. By contrast, if the 3D $CSO(p,F_1)$ and $CSO(p,F_2)$, shown by green shadow and blue shadow respectively, are united, all space is able to accommodate the instrument apart from the shadowed area. One can conclude that it is mandatory to compute the union of all 3D *CSOs* rather than only the spherical projected images in order to achieve accurate analysis result.

Step 4. Compute *LICS(p,F)* and $GICS(p,P_s)$.

LICS(p,F) is calculated as

$$LICS(p,F) = \{d \in MS(p,I): d \cdot n \geq 0\} \quad \text{(Eq6.15)}$$

where n is the outward normal of face F at point p. As all surfaces of the object are triangulated, the target point **p** must be on one triangular facet F. The $LICS(p,F)$ is therefore a hemisphere located at **p** with the radius R, whose pole is at the direction of the outward normal of F.

$GICS(p,P_s)$ is then calculated as:

$$GICS(p,P_s) = LICS(p,F) - \bigcup_{O_i \in P_s} CSO(p,O_i) \quad \text{(Eq6.16)}$$

The visibility of the target point **p** is assured as long as the specified instrument is located in the derived $GICS(p,P_s)$.

6.2.5 Computation of visibility cones

It is clear that the above algorithm leads to the exact solution space of the task if the computational operations are handled correctly and efficiently. Nevertheless, numerical accuracy and computational complexity are two indispensable issues existing in geometric computation. Aiming to cope with the challenging geometry computations involved throughout the algorithm such as spherical projection and Boolean operations, this research takes the advantage of powerful rendering techniques and graphic engine available in modern CAD software. Both spherical projection and 3D Boolean operations can be performed efficiently with appropriate pre-process preparation. Despite the approximate computation result, it satisfies the requirements of inspection planning.

6.2.5.1 Spherical projection

As far as the unit sphere is concerned, the spherical projection of a point set P_s is calculated as

$$\phi(p) = \left\{\frac{\overline{cp}}{|cp|} \,|\, p \in P_s\right\} \quad \text{(Eq6.17)}$$

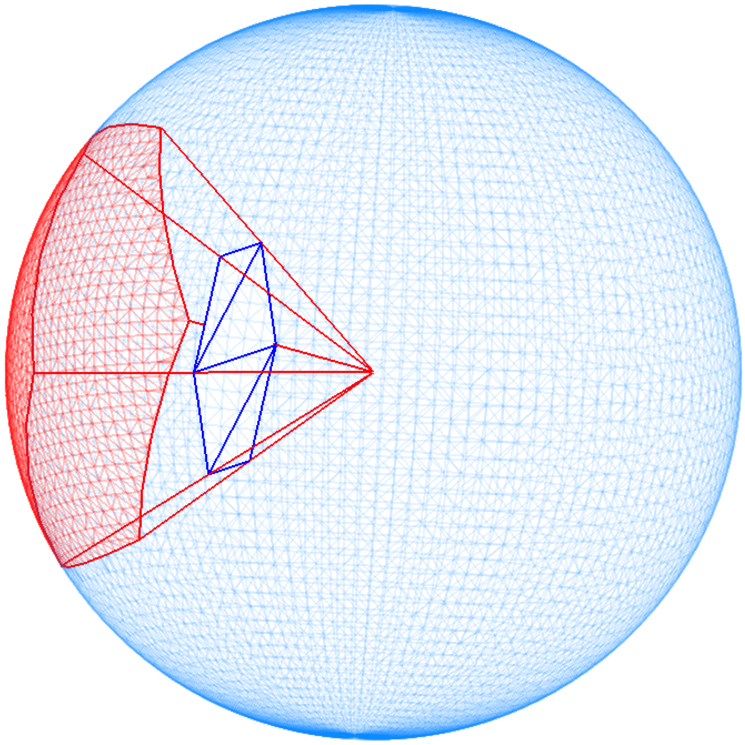
where **p** is any point in the set and **c** is the origin of the unit sphere.

One important property of the spherical projection is that the spherical projection of a closed point set P_s is equal to the union of the spherical projection of its boundary (Liu et al., 2009). It can be further extended as, for a polyhedral surface$S = \{F_1, F_2, \dots, F_n\}$, the spherical projection of S can be computed from the border curves of facets that consists its boundary. The above properties valid for spherical projection to sphere with any radius since the function of the projection process can be written as:

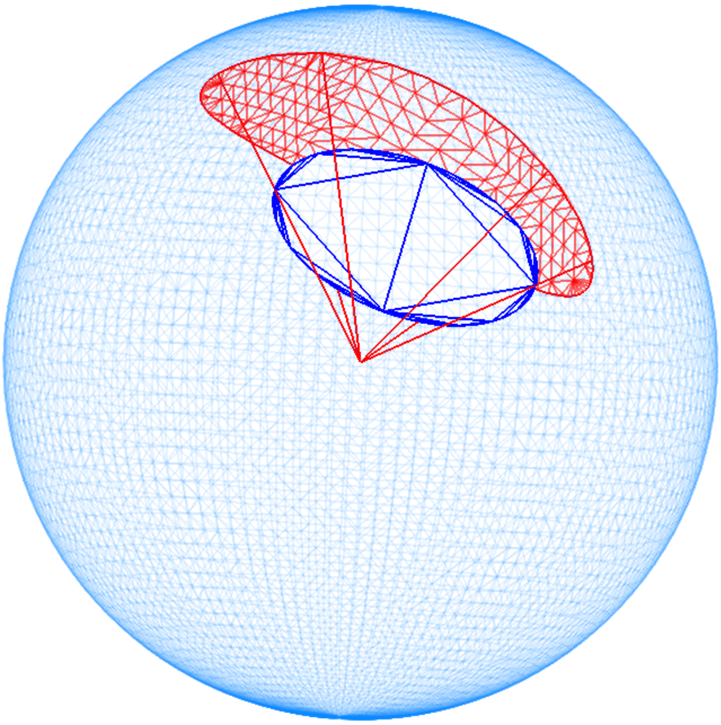
$$\phi(p) = \left\{\left(\frac{\overline{cp}}{|cp|}\right) R | p \in P_s\right\} \quad \text{(Eq6.18)}$$

where R is the radius of the sphere that supports the projection.

The search for boundary of a surface can be carried out within CAD software efficiently and the result is spherically projected to the designated sphere afterwards. Figure 6.23 illustrates this property of the spherical projections in a CAD environment. The normal-facing hexagonal surface in Figure 6.23(a) is consisted of four triangular facets in blue and the central projection is carried out on the sphere shown in blue meshes, resulting in the projected area denoted by the red mesh. Instead of projecting all edges of the facets, only edges on the border curves of the surface were projected to obtain the boundary projection. Figure 6.23(b) shows the projection of a non-polyhedral surface to a sphere. The ellipsoidal surface is not normal-facing the target and is triangulated first. By means of only projecting the bounding edges of triangular facets along the boundary of the surface, the process is simplified tremendously compared with projecting all facets on the sphere. Note that in both figures the projecting lines shown in red are only for better illustrative results.



(a)



(b)

Figure 6.23 (a) Spherical projection of a triangulated hexagonal surface to a sphere

(b) Spherical projection of a triangulated ellipsoidal surface to a sphere

6.2.5.2 Boolean operations

Minkowski sum is chosen to perform the Boolean operation such as union and intersection as the main principle to calculate the exact intersection of visibility cones since (Spyridi and Requicha, 1990; Lim and Menq, 1994; Liu et al., 2009). Implementations for both convex objects and general polyhedral are available (Varadhan and Manocha, 2006; Fogel and Halperin, 2007). This has been demonstrated as a computationally intensive process that increases enormously when more elements are introduced into the 3D environment. Some researchers applied different strategies to avoid or simplify the Minkowski sum operation using approximate methods e.g. convoluting the boundary of two polyhedral (Guibas and Seidel, 1986; Basch et al., 1996) and generating a subset of the Minkowski sum that shares the same spherical projection (Liu et al., 2009). Nevertheless, the complexity of the computation is still the major drawback that severely limits its application.

Benefiting from the powerful graphic cards available in personal computers nowadays, 3D Boolean operations can be conducted efficiently within CAD environment with acceptable accuracy level. This is the underpinning technique of the computation methodology adopted in the algorithm.

Boolean operations including intersection, union and difference are default commands supplied by CAD software. All involved computation elements such as $MS(p,I)$, $CSO(p,O)$, $LICS(p,F)$ and $GICS(p,P_s)$ are stored as geometrical entities in the library once obtained, which can be performed all Boolean operations as needed.

6.2.5.3 Algorithm example

Three examples are given in this section to demonstrate the *ICS* algorithm.

(a) The object is a pad and two target points are located on two surfaces of the pad, shown in Figure 6.24(a). All surfaces of the object have been triangulated in Figure 6.24(b) and 20 facets are obtained in this case. The algorithm of computing *ICS* is then carried out for both points respectively.

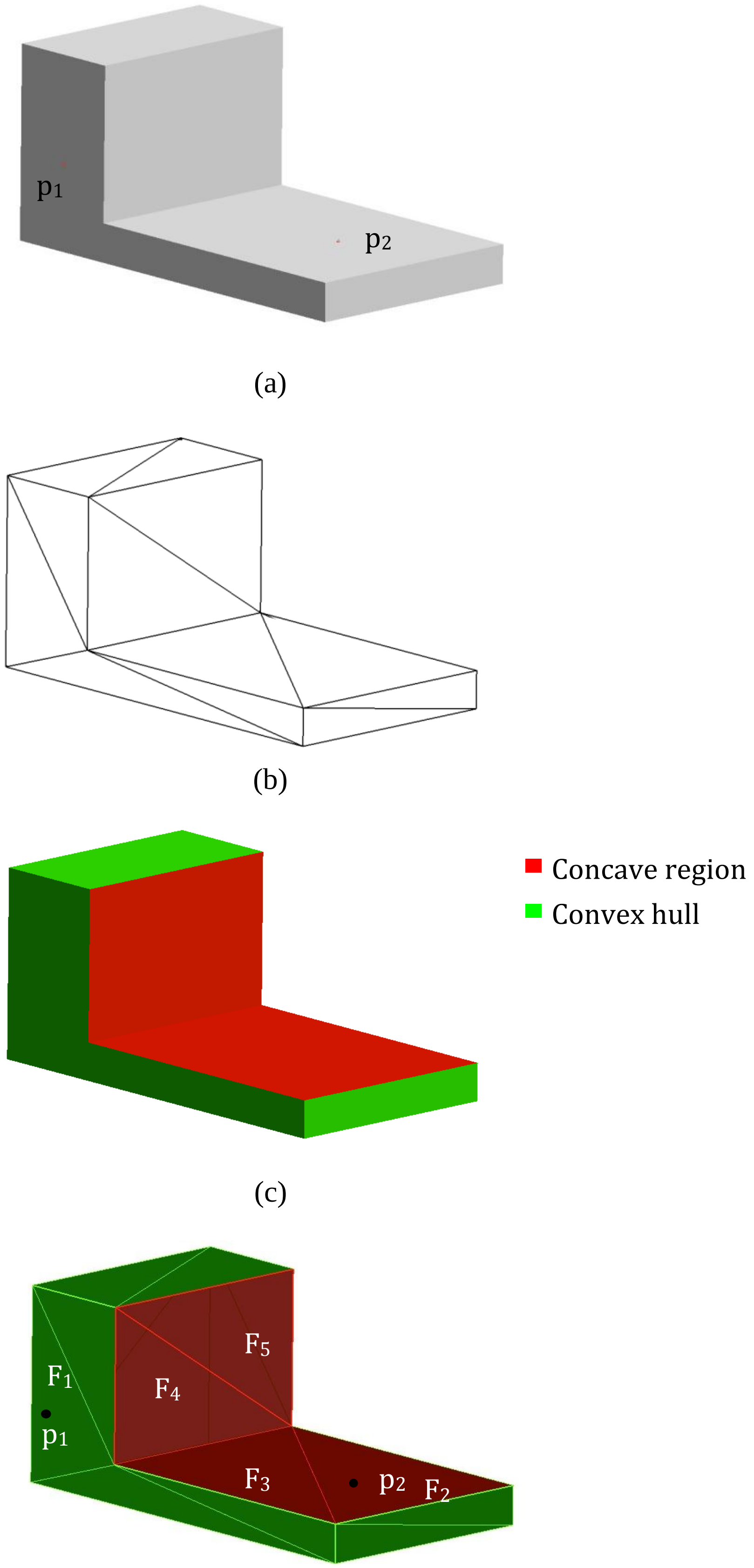
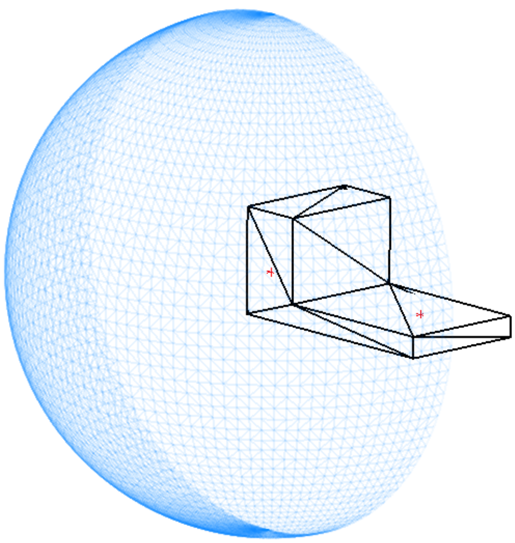


Figure 6.24 Example 1 of visibility analysis

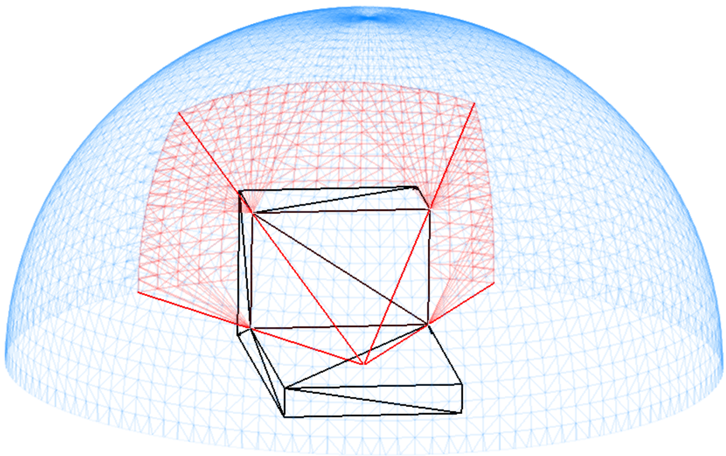
Step 1. Previous stage of selecting instrument has determined that laser tracker is the suitable inspection system and *MS(p,I)* is constructed accordingly.

Step 2. The convex hull of the object is calculated and surfaces are classified into hull facets and non-hull facets. Figure 6.24(c) and (d) indicate the two groups of facets using different colours, red for concave region and green for convex hull. As far as $\mathbf{p_1}$ is concerned, no other facets can affect its visibility apart from F_1, the facet on which it locates since it is on a hull facet according to Property 1. Therefore $OFS(p_1,O)$ only contains the carrying facet of the target point. On the contrary, $\mathbf{p_2}$ is on a non-hull facet F_2 in the concave region of the object. Neighbouring facet F_3 shares a convex edge with F_2 and hence it is not regarded as neighbouring occluder. Facets F_4 and F_5 are deemed as potential occluders based on the criteria: (1) the facets are involved in the same concave region; (2) the facets are not coplanar; (3) the facets are both presented fully or partially in each other's positive space. Consequently, $OFS(p_2,O)$ contains facets F_2, F_4 and F_5.

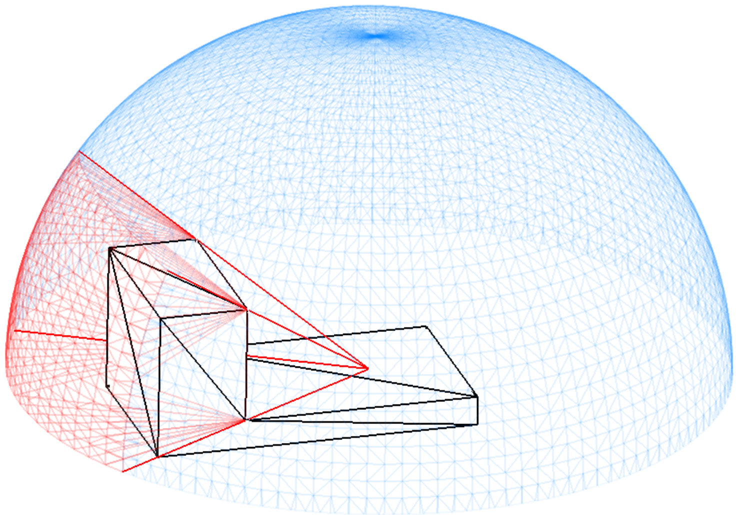
Step 3. *CSO(p,O)* is then constructed by projecting each occluder on the sphere defined by *MS(p,I)*. Since the instrument is viewable to **p** only when it is located in the positive half space of **p**, it is preferred to first compute the part of *MS(p,I)* in the viewable space of **p**, which is the hemisphere originated at **p** according to the carrying facet. Subsequent projection and Boolean operation are carried out in the confined hemisphere avoiding redundant computation in the negative half space of **p**.



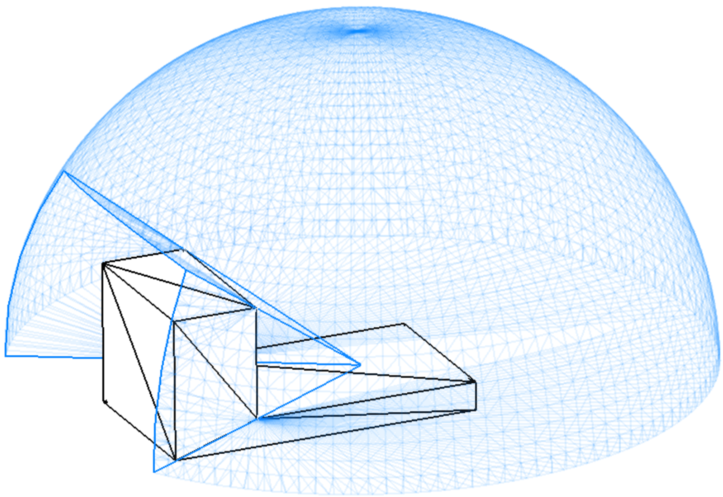
(a)



(b)



(c)



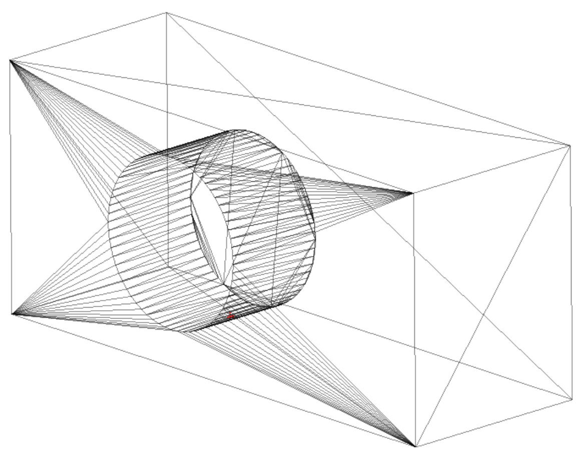
(d)

Figure 6.25 Process of calculating the GICS

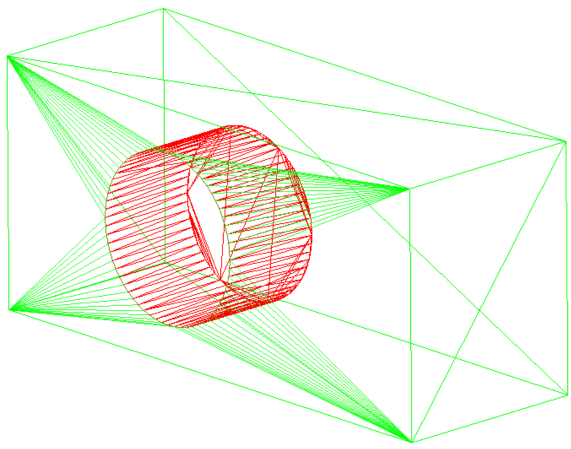
Figure 6.25(a) illustrates the constructed hemisphere for $\mathbf{p_1}$ using F_1 and $CSO(p_1,O)$ is unavailable as $OFS(p_1,O)$ is empty after eliminating F_1 from the occluder set. $CSO(p_2,O)$ is created by projecting F_4 and F_5 to the hemisphere based on F_1 and the result is shown by red mesh in Figure 6.25(b) and (c). Note that only the border curves of F_4 and F_5 are projected in the process to save computation.

Step 4. $LICS(p,F)$ is effectively the previously obtained hemisphere with the origin at **p** due to the fact that the hemisphere is the *ICS* if only the carrying facet is taken into account. $GICS(p_1,P_s)$ is equivalent to $LICS(p_1,F)$ since there is no global occluder obstructing visibility. $GICS(p_2,P_s)$ is generated eventually using the difference Boolean operation between $LICS(p_2,F)$ and $CSO(p_2,O)$ and represented by the blue meshed space in Figure 6.25(d). Visibility of point $\mathbf{p_2}$ is guaranteed once the selected instrument is allocated within the bounded space.

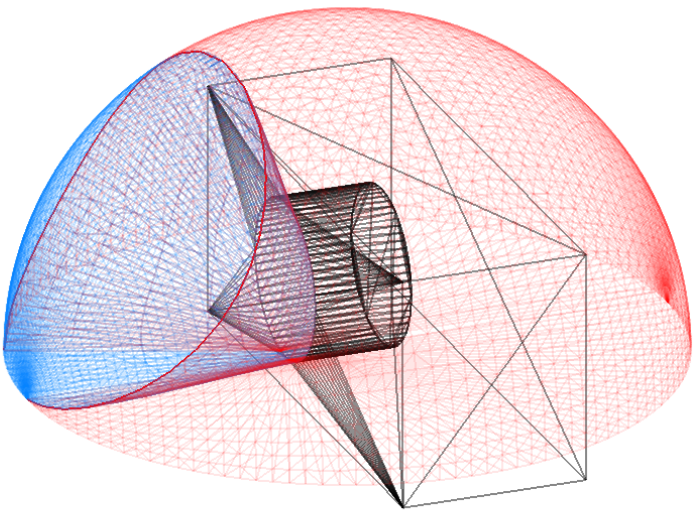
(b) The object is a pad with a blind hole and the target point is located on the surface in the hole, shown in Figure 6.26(a). All surfaces of the object have been triangulated and thus curve surfaces within the hole are converted into planar facets. Hull facets are accordingly computed coloured in green while non-hull facets are shown in red in Figure 6.26(b). Since all non-hull facets are in the same concave region with **p** and regarded as occluders based on the criteria, the border curves of the blind hole are projected to $LICS(p,F)$ resulting in $CSO(p,O)$ shown by red meshes in Figure 6.26(c). $GICS(p,P_s)$ is therefore constructed by subtracting $CSO(p,O)$ from $LICS(p,F)$ indicated by the blue mesh space, which is configurable for the selected instrument.



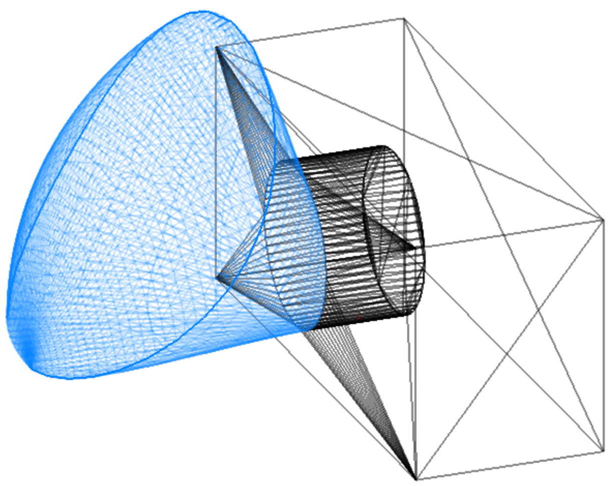
(a)



(b)



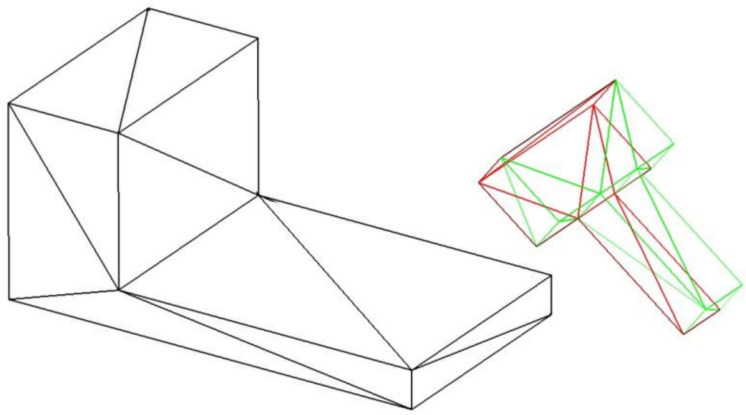
(c)



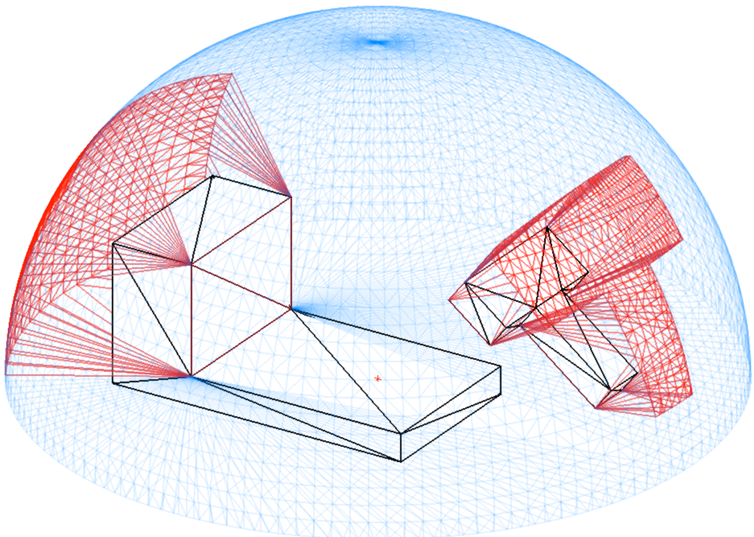
(d)

Figure 6.26 Example 2 of visibility analysis

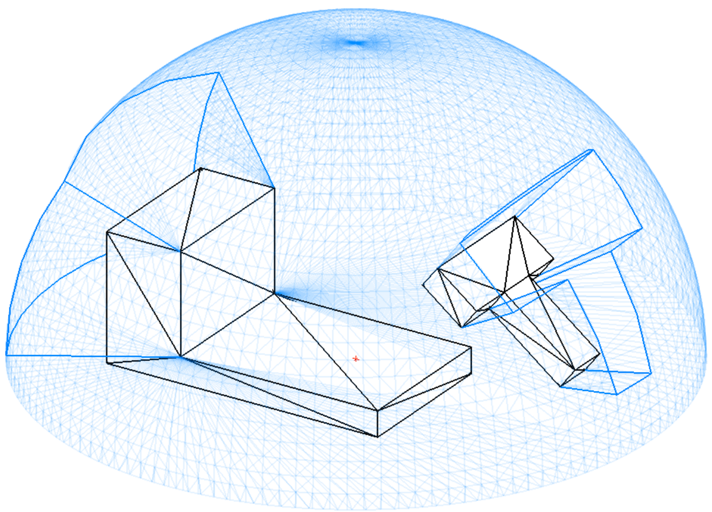
(c) Example 3 illustrates the case that surrounding object is considered in addition to the target object. The same object in Example 1 is analysed but with two additional bounding boxes representing other obstacles such as gigs and other instruments in the working environment. It is not necessary to compute the hull facets and non-hull facets for the bounding box since Property 1 is not valid once the target point is not located on the boundary surfaces of the object. Under this circumstance, only visibility culling techniques can be applied. The facets that affect the visibility of **p** are shown in red after back-face culling is conducted and all green facets are ignored due to the fact that **p** is back facing to them.



(a)



(b)



(c)

Figure 6.27 Example 3 of visibility analysis

The occluded facets are projected on *LICS(p,F)* to obtain *CSO(p,O)* in Figure 6.27 (b) and *GICS(p,*P_s*)* is generated afterwards by subtracting *CSO(p,O)* from *LICS(p,F)* indicated by the blue mesh space in Figure 6.27(c). Benefiting from the 3D Boolean operation rather than spherical image intersection, space from **p** to the occluded surfaces is retained where the instrument can still be configured.

6.3 Summary

In order to improve the effectiveness and efficiency of the inspection planning processes, it is imperative to consider the accessibility domain of measurement instruments. This is a prerequisite to the next stage, namely clustering analysis. Accessibility and visibility analyses have been widely applied in manufacturing processes such as injection moulding, NC machining, and CMMs inspection path planning, whereas research related to newer instruments like laser trackers is limited.

This chapter presents a methodology for developing a visibility analysis system applied for LVM systems such as laser tracker. The local visibility cone (LVC) and global visibility cone (GVC) for discrete measurement points are generated respectively. The common GVCs are then derived by computing the intersections of

GVCs of all sampled points and measurement clusters are defined based on the results.

This research is the first attempt to carry out visibility analysis for point-based LVM instruments. Based on the methodologies developed for accessibility analysis for CMMs, computational graphic techniques as well as 3D Boolean operations provided by modern CAD were adopted to develop the proposed approach to solve the computational intensive Minkowski sum. Visibility cones are output subsequently to the following processes for determining the optimised configuration of the instrument.

CHAPTER 7 CLUSTERING ANALYSIS

7.1 Introduction

From the previous visibility analysis module, the output is a set of GICSs for all inspection points, which are presented by different spatial regions in the digital environment. It is neither optimistic nor realistic to locate the instrument throughout all GICSs in order to accomplish the inspection task. Extensive research efforts have been found in the literature attempting to cluster the inspection operations on CMMs based on three criteria: (1) set-up changes; (2) probe changes; (3) probe orientation changes (Lim and Menq, 1994; Limaiem and ElMaraghy, 1999; Zhang et al., 2000; Vafaeesefat and ElMaraghy, 2000a; Hwang et al., 2004). Minimizing those resources results in the optimized solution of clustering. As far as most LVM instruments are concerned, changing the set-up of a specified instrument, e.g. location, is time consuming since recalibration is mandatory. Moreover, changing the location of the instrument means either bundling or best-fitting the new coordinates into a common coordinate system where the complexity of the coordinate uncertainty for each point increases significantly. It also introduces additional uncertainty due to the alteration of temperature gradient from the station to the target and correlating compensation for instruments whose performance varies with ambient temperature. Under this circumstance, inspecting all of the given points with a minimum number of relocations of the instrument is the primary constraint while clustering the measurement operations. According to the definition of GICS, two points $\mathbf{p}_1$ and $\mathbf{p}_2$ can be measured at the same instrument location where their corresponding GICSs intersect. From the previous visibility analysis, GICS is determined for each inspection point and presented by a spatial region in digital environment. With the purpose of reducing the number of location changes of the instrument, GICSs must be clustered and intersected in such a manner that the consolidated GICSs are visible to the greatest number of inspection points.

7.2 Proposed clustering algorithm

Since the obtained GICSs from visibility analysis are continuous spatial regions, clustering techniques proposed for CMMs and VMs are incapable of finding the appropriate clusters that cover all the GICSs. A new approach must be developed aiming to consolidate and group all GICSs into clusters that cover all inspection points while maximum coverage of each cluster is achieved.

7.2.1 Set covering problem

Set covering problem (SCP) is a well-known computational problem and proven to be NP-complete (Garey and Johnson, 1979). It is the mathematical model of a variety of optimization problems such as edge covering, vertex cover, set partitioning and set packing (Haouari and Chaouachi, 2002). It also underpins several practical resource allocation problems including configuring the wireless network coverage (Shorey, 2006), selecting the necessary sentences in a speech recognition model (Nemhauser and Wolsey, 1988), airline crew scheduling (Hoffman and Padberg, 1993), vehicle routing (Desrosiers et al., 1995), facility location (Current and O'Kelly, 1992), and the IBM virus identification (Williamson, 1998).

Given a finite set $A = \{a_1, \cdots, a_n\}$, known as the universe, and subsets $S = \{S_1, \cdots, S_m\} \subseteq A$, every element in the universe *A* is covered by at least one subset from *S* and the union *C* of those subsets forms the entire universe, namely $\bigcup_{S_i \epsilon C} S_i = A$. The cost of the set covering is the size of the subsets union$|C|$, which indicates the number of subsets it includes. The aim is to find the *C* whose size is minimal, namely the cost of this set covering is minimal.

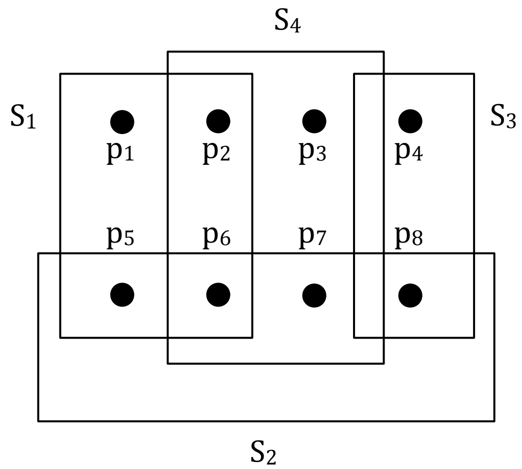


Figure 7.1 An instance of the minimum set covering problem

An instance is given in Figure 7.1. There are 8 elements in the universe $A = \{p_1, \cdots, p_8\}$ and a set of subsets of A are $S = \{S_1, S_2, S_3, S_4\}$. A minimum size set cover is $C_1 = \{S_1, S_3, S_4\}$, which has the size of 3. A set cover $C_2 = \{S_1, S_2, S_3, S_4\}$ includes all the elements in the universe as well although a size of 4 is achieved.

The set-covering problem can also be described as an integer program. Given a zero-one matrix (a_{ij}) consisted of m-row and n-column, define

$$\begin{cases} x_j = 1 & \text{if column j is in the solution} \\ x_j = 0 & \text{otherwise} \end{cases}$$

The SCP is to

$$\text{Minimize } \sum_{j=1}^{n} c_j x_j \qquad \text{(Eq7.1)}$$

where c_j is the cost and $c_j > 0$, subject to

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1, i = 1, \cdots, m \qquad \text{(Eq7.2)}$$

$$x_j = 0 \text{ or } 1, j = 1, \cdots, n \qquad \text{(Eq7.3)}$$

Eq7.2 guarantees that there is at least one column covering each row while Eq7.3 maintains the binary value of the elements in the matrix. Therefore the SCP is the problem of finding a subset of columns of a zero-one matrix such that all rows are covered by at least one column in this subset.

The example shown above can be converted to the following process:

(1) All elements in the universe A construct the rows $\{p_1, \cdots, p_8\}$ in the matrix and the columns are consisted of all the subsets $S = \{S_1,\ S_2,\ S_3, S_4\}$. Binary values are then assigned to each element of the matrix according to the relation shown in Figure7.3 and the generated matrix is then presented as:

$$
\begin{array}{c|cccc|}
 & S_1 & S_2 & S_3 & S_4 \\
p_1 & 1 & 0 & 0 & 0 \\
p_2 & 1 & 0 & 0 & 1 \\
p_3 & 0 & 0 & 0 & 1 \\
p_4 & 0 & 0 & 1 & 0 \\
p_5 & 1 & 1 & 0 & 0 \\
p_6 & 1 & 1 & 0 & 1 \\
p_7 & 0 & 1 & 0 & 1 \\
p_8 & 0 & 1 & 1 & 0
\end{array}
$$

(2) Find the minimum subset C of columns $S = \{S_1,\ S_2,\ S_3, S_4\}$ that has the value of 1 with every row $A = \{p_1, \cdots, p_8\}$ resulting in $C_1 = \{S_1, S_3, S_4\}$, which satisfies both Eq7.2 and Eq7.3.

Having recalled the objective of GICSs clustering, it is clear that this is the process that allocates resources (GICSs) to a set of targets (inspection points) where minimum cost can be achieved, specifically minimum involved GICSs. Therefore the minimum set covering problem should be able to provide the mathematical model and solution to the clustering algorithm.

7.2.2 Clustering process modelling

The modelling process is given as follows:

1) Construct the rows of the matrix, namely the universe A, which are consisted of all inspection points $\{\boldsymbol{p_1}, \cdots, \boldsymbol{p_i}\}$.

2) Construct the columns, which are the subsets $\boldsymbol{S} = \{\boldsymbol{S_1}, \cdots, \boldsymbol{S_j}\}$ obtained by intersecting individual GICS.

An iterative intersecting process is conducted at the beginning between every GICS against the others. The details of this intersection identification process will be revealed later in this chapter. As a result, there are two types of individual GICS existing in a normal situation: GICS intersecting with other GICS/GICSs, and GICS intersecting with no other GICS/GICSs. For isolated GICS, the corresponding subset S only includes its own inspection point. For intersecting GICSs, the corresponding subset S comprises all related inspection points. Under this circumstance, every inspection point is covered by S since it is at least included in its own GICS and one inspection point can be included in different subsets since its GICS can intersect with multiple GICSs. The columns of the matrix are constructed accordingly using subsets $S = \{S_1, \cdots, S_j\}$.

3) Assign binary value to the elements of the matrix.

4) Find the minimum size set cover $\boldsymbol{C} = \{\boldsymbol{S_1}, \cdots, \boldsymbol{S_k}\}$, which meets the condition that for every row there is at least one selected column such that $a_{ij} = 1$.

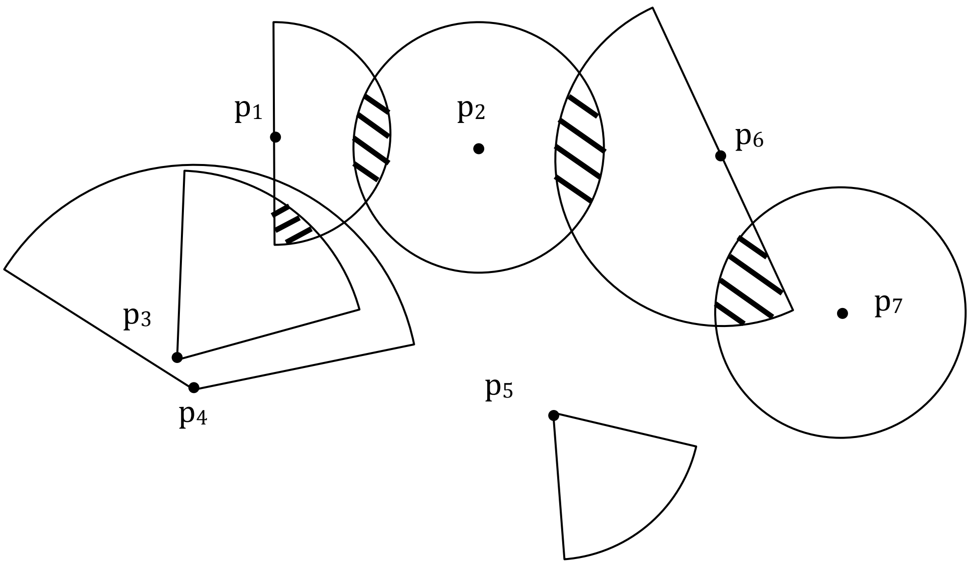


Figure 7.2 An example of clustering process using 2D GICSs

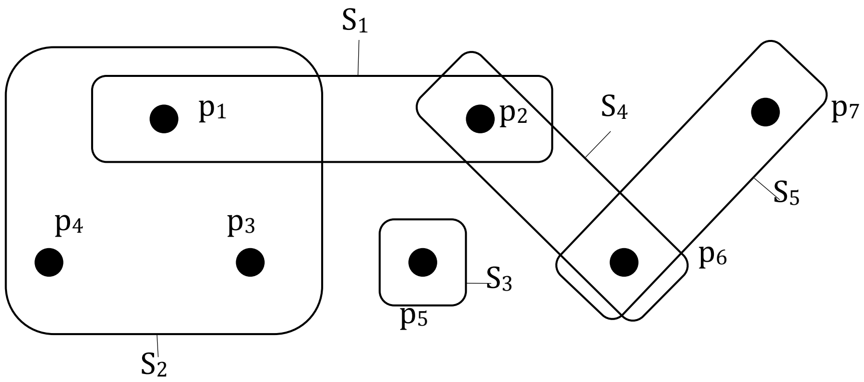


Figure 7.3 Set covering presentation of the example

A 2D instance of a typical distribution of GICSs is shown in Figure 7.2. Note that multiple instruments are involved therefore the diameters of GICSs differentiate from each other. It can be seen that *GICS(p_1)* is intersecting with *GICS(p_2)*, *GICS(p_3)*, and *GICS(p_4)* and *GICS(p_2)* is intersecting with *GICS(p_6)* that intersects with *GICS(p_7)* at the same time. *GICS(p_5)* is intersecting with on other GICS. The equivalent set covering version of this specific problem is given in Figure 7.3. Every inspection point is the element in the universe $A = \{p_1, \cdots, p_7\}$ and the set of subsets $S = \{S_1,\ S_2,\ S_3, S_4, S_5\}$ is consisted of each intersection among the GICSs except S_3 that intersects no other GICS. The corresponding binary matrix is then constructed as:

	S_1	S_2	S_3	S_4	S_5
p_1	1	1	0	0	0
p_2	1	0	0	1	0
p_3	0	1	0	0	0
p_4	0	1	0	0	0
p_5	0	0	1	0	0
p_6	0	0	0	1	1
p_7	0	0	0	0	1

The minimum set cover can then be easily obtained as $C_1 = \{S_2, S_3, S_4, S_5\}$, where $\{p_1, p_3, p_4\} \epsilon\ S_2$, $\{p_5\} \epsilon\ S_3$, $\{p_2, p_6\} \epsilon\ S_4$, and $\{p_6, p_7\} \epsilon\ S_5$. One may notice that in this particular example there is another minimum set cover $C_2 = \{S_1, S_2, S_3, S_5\}$. This is because p_1 can be inspected within either S_1, or S_4 and p_6 can be inspected within either S_4, or S_5. No further analysis will be given at this stage since the alternatives will be evaluated based on the performance in terms of uncertainty later on.

7.2.3 Solution to SCP

In the previous example, the minimum covering set can be obtained using intuitional heuristics. Nevertheless the SCP is well known as notoriously hard to solve with an optimum solution (Chvatal, 1979). Extensive research efforts have been made attempting to solve this NP-complete problem by means of exact approach and approximate approach, respectively. Methods based on dual heuristic (Fisher and P. Kedia, 1990), Lagrangian heuristic (Beasley, 1987; Beasley and Jornsten, 1992) and column subtraction are capable of obtaining exact solution to instances with a few hundred rows and a few thousand columns using a considerable amount of computational resources (Harche and Thompson, 1994). In order to cope with practical problems which often include much larger instances, the research interest shifts to solve the SCP approximately using heuristic approaches such as Lagrangian relaxation (Beasley, 1990), genetic algorithm (Beasley and Chu, 1996), simulated annealing (Sen, 1993; Brusco et al., 1999), neural network (Jeffries, 1991), and greedy heuristic (Chvatal, 1979). A comparative review of those approximate approaches was given by Grossman and Wool (1994) based on a set of random-generated problems with up to 500 rows and 5000 columns as well as two specific combinatorial questions with up to 28160 rows and 11264 columns. The results indicated that although neural network algorithm performed the best in random problems with significant execution time, greedy heuristic and its alternatives generated very close solution with a remarkable faster execution speed whereas other techniques performed inferiorly (Grossman and Wool, 1994). The greedy heuristic algorithm is consequently chosen as the approximate approach to clustering the GICSs due to its simplicity and yet adequate accurate result. The details of other algorithms are beyond the scope of this thesis and the audiences are suggested to refer the mentioned literature for the comprehensive introduction.

Aiming to find the locally optimal choice at each step, the greedy algorithm is a well-applied optimization algorithm that can generate optimal solution to problems such as finding minimal spanning trees, finding single source shortest paths and finding optimum Huffman trees (Wikipedia) while no optimal solution can be obtained for most NP-complete problems but a global optimal approximation is outputted. Greedy heuristic was first adopted by (Johnson, 1974; Lovász, 1975) as the solution to unicost SCP and Chvatal (1979) applied this approximate algorithm to weighted SCP. The algorithm can be state as the following process:

Recall zero-one matrix (a_{ij}), C is the cover; c_j is the cost of each column and a_j denotes each column. Therefore $|a_j|/c_j$ represents the number of points covered by a_j per unit cost.

Step 0. Set $C = \emptyset$.

Step 1. If a_{ij} = 0 then stop: C is the cover. Otherwise find the subscript k that maximize the ratio $|a_j|/c_j$, and then go Step 2.

Step 2. Add k to C and replace each column a_j by a_j -a_k. Return to Step 1.

In this clustering problem, each GICS is considered equally and as a result the cost of each column is the same. The process of finding the minimal set cover is as following: (1) The algorithm initializes an empty set C and later obtained cover is included to C. (2) Select the column containing maximum number of elements that have the value of 1. This column is subsequently added to C set as the first element. (3) Assign value 0 to a_{ij}, which has been covered in the previous step. (4) Return to the second step until the matrix (a_{ij}) = 0. The algorithm is as follows:

Step 0. Set $C = \emptyset$.

Step 1. While $(a_{ij}) \neq 0$,

do select a column a_k that has maximum $|a_j|$,

$a_j = a_j - a_i$,

add k to C.

Step 3. Return C.

The greedy algorithm is demonstrated as following. Recall the example shown in Figure 7.2 and the corresponding matrix is

	S_1	S_2	S_3	S_4	S_5
p_1	1	1	0	0	0
p_2	1	0	0	1	0
p_3	0	1	0	0	0
p_4	0	1	0	0	0
p_5	0	0	1	0	0
p_6	0	0	0	1	1
p_7	0	0	0	0	1

Let $C = \emptyset$ and column S_2 is selected as it has the longest size, namely containing the largest number of "1". Eq7.2 enforces S_3, which covers p_5 exclusively, to be added to C in order to maintain the condition that every row should be covered. The matrix is updated by assigning the value of 0 to every row covered by S_2 and S_3 already resulting in $C = \{S_2, S_3\}$. Therefore all elements in row p_1, p_2, p_3 and p_5 are set to null, shown in grey:

	S_1	S_2	S_3	S_4	S_5
p_1	0	0	0	0	0
p_2	1	0	0	1	0
p_3	0	0	0	0	0
p_4	0	0	0	0	0
p_5	0	0	0	0	0
p_6	0	0	0	1	1
p_7	0	0	0	0	1

There are two columns S_4 and S_5 that both have the size of 2 and the algorithm considers as the tie situation. S_4 is selected first and S_5 will be chosen as the second possible cover. If S_4 is selected and then added to C, the matrix is

$$\begin{array}{c|ccccc|}
 & S_1 & S_2 & S_3 & S_4 & S_5 \\
p_1 & 0 & 0 & 0 & 0 & 0 \\
p_2 & 0 & 0 & 0 & 0 & 0 \\
p_3 & 0 & 0 & 0 & 0 & 0 \\
p_4 & 0 & 0 & 0 & 0 & 0 \\
p_5 & 0 & 0 & 0 & 0 & 0 \\
p_6 & 0 & 0 & 0 & 0 & 0 \\
p_7 & 0 & 0 & 0 & 0 & 1
\end{array}$$

It is clear that in order to obtain a complete cover S_5 must be added to C and the matrix then becomes zero. The minimum set is $C = \{S_2, S_3, S_4, S_5\}$.

If S_5 is selected in the previous step, the matrix then becomes:

$$\begin{array}{c|ccccc|}
 & S_1 & S_2 & S_3 & S_4 & S_5 \\
p_1 & 0 & 0 & 0 & 0 & 0 \\
p_2 & 1 & 0 & 0 & 1 & 0 \\
p_3 & 0 & 0 & 0 & 0 & 0 \\
p_4 & 0 & 0 & 0 & 0 & 0 \\
p_5 & 0 & 0 & 0 & 0 & 0 \\
p_6 & 0 & 0 & 0 & 0 & 0 \\
p_7 & 0 & 0 & 0 & 0 & 0
\end{array}$$

An alternative situation occurs since the remaining columns S_1 and S_4 have the same non-zero element covered. The corresponding cover sets are $C = \{S_1, S_2, S_3, S_5\}$ and $C = \{S_2, S_3, S_4, S_5\}$ and one can find that the latter is the same cover set when S_5 was selected. The minimum covers obtained using greedy algorithm are consistent with the results from the example given in Figure 7.3, which are the optimal solutions to this specific problem.

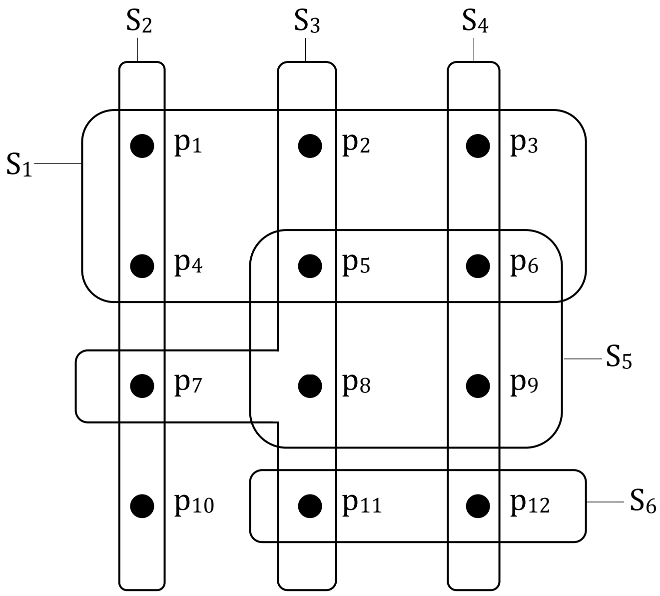


Figure 7.4 Example of where non-optimal solution is obtained

Another instance is given in Figure 7.4 where non-optimal solution is obtained. The example includes 12 inspection points $A = \{p_1, \cdots, p_{12}\}$ and the intersections among GICSs have been computed indicated by $S = \{S_1,\ S_2,\ S_3, S_4, S_5, S_6\}$. It can be seen that:

$S_1 = \{p_1,\ p_2,\ p_3, p_4, p_5, p_6\}$,

$S_2 = \{p_1,\ p_4,\ p_7, p_{10}\}$,

$S_3 = \{p_2,\ p_5,\ p_7,\ p_8, p_{11}\}$,

$S_4 = \{p_3,\ p_6,\ p_9, p_{12}\}$,

$S_5 = \{p_5,\ p_6,\ p_8, p_9\}$,

$S_6 = \{p_{10},\ p_{11},\ p_{12}\}$

The corresponding matrix is then constructed as:

	S_1	S_2	S_3	S_4	S_5	S_6
p_1	1	1	0	0	0	0
p_2	1	0	1	0	0	0
p_3	1	0	0	1	0	0
p_4	1	1	0	0	0	0
p_5	1	0	1	0	1	0
p_6	1	0	0	1	1	0
p_7	0	1	1	0	0	0
p_8	0	0	1	0	1	0
p_9	0	0	0	1	1	0
p_{10}	0	1	0	0	0	1
p_{11}	0	0	1	0	0	1
p_{12}	0	0	0	1	0	1

Following the greedy algorithm, S_1 is picked initially since it covers the maximum number of inspection points. S_3 is added to the cover set C afterwards since it covers 3 elements, which is the highest in the remaining. S_4 and S_2 are then selected by the algorithm sequentially as they cover 2 and 1 uncover elements respectively. The minimum cover set is then created as $C = \{S_1, S_2, S_3, S_4\}$. However, this is not the optimal solution to the problem since a smaller cover set can be found, which is $C = \{S_2, S_3, S_4\}$. The algorithm missed the chance of identifying the optimal solution at the beginning by selecting the set that covers the maximum number of uncovered elements.

The adopted greedy heuristic algorithm has been proved by Chvatal (1979) that has an approximate factor *H(m)* to the optimal solution at the worst case, where *m* is the largest set size and $H(m) = \sum_{i=1}^{m} 1/m \approx \ln(\mathrm{m})$.

Throughout the experimentations conducted by Grossman and Wool (1994), the greedy algorithm achieved an average deviation of 3.10% to the optimal solution. This approximate solution is considered adequate to the clustering problem and the outstanding computation speed is preferable as well as the relative simple data structure while a large number of inspection points are involved. Nevertheless more accurate approximate algorithm such as the neural network algorithm or even exact algorithm is desirable to generate an optimal cluster in future research.

7.2.4 Intersection of the GICSs

As mentioned in previous section, the subset *S* of the universe is consisted of intersections among all involved GICSs. An iterative intersecting process is applied to check the relation between each GICS against the others. As the GICS is represented using entity in the CAD environment, the intersection check can then by carried out using the following algorithm:

N is the number of involved GICSs and *p* is the corresponding inspection point. *S* is the subset of the GICSs and *C* is the cover set.

Step 0. set $i=0, j=1$.

Step 1. $i=i+1$. Add $S_i = \{p_i\}$ to *C*.

Step 2. $j=j+1$

Step 3. CHECK_COLLSION between *GICS(i)* and *GICS(j)*

If *True* THEN compute the intersection and update the generated entity to

GICS(i). Update $S_i = \{S_i, p_j\}$, goto Step 2 while $j > N$

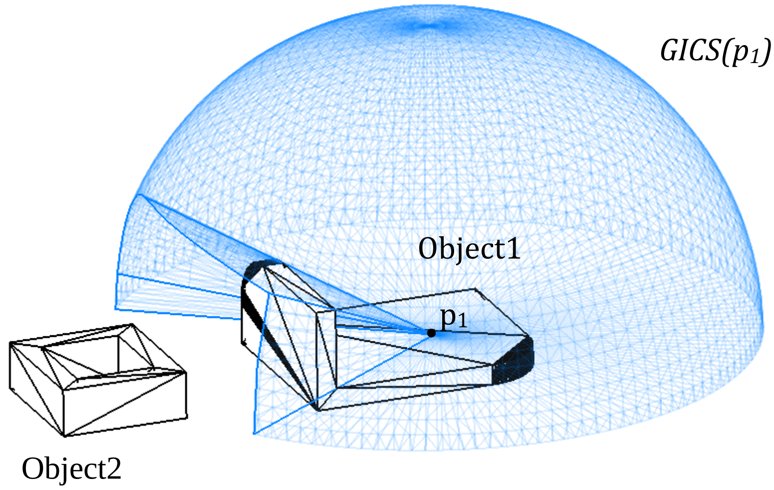
ELSE goto Step 2 until $j > N$

Step 4. goto Step 1 until $i > N$

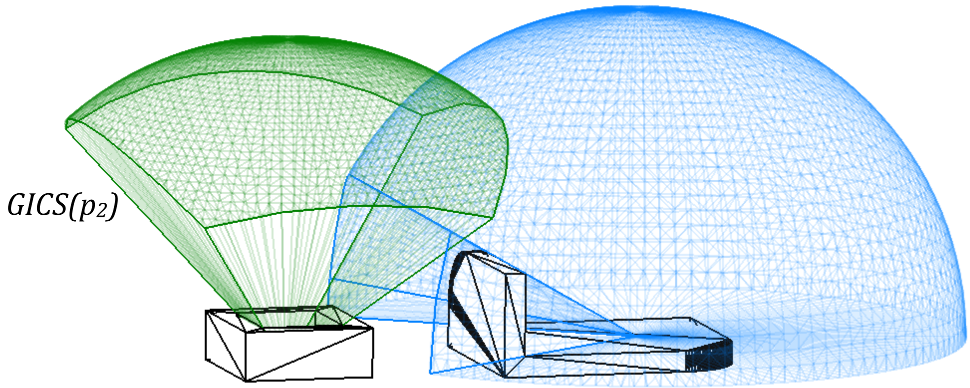
Note that the cover set *C* here is not the minimum set cover but a set that comprises all intersections among GICSs. As mentioned in previous chapter, this research adopted the 3D Boolean operations supported by CAD software, CATIA in this case. Therefore the intersection process can be conducted within CAD environment efficiently with an adequate accuracy. The command CHECK_COLLSION is a typical function available in the function library for user to call. A binary value is outputted depending on if there is intersection existing between two entities. This function involves considerably less computational forces compared with the Boolean operations and consequently consumes less time and cost.

This algorithm can be demonstrated using the example given in Figure 7.2. *GICS(p_1)* is firstly include in C and the intersection between *GICS(p_1)* and *GICS(p_2)* is identified and computed shown in the crossed area without any further intersection with other entities. Therefore $S_1 = \{p_1, p_2\}$ is added to the cover set $C = \{S_1\}$. Intersection between *GICS(p_1)* and *GICS(p_3)* is then found and the generated entity also intersects with *GICS(p_4)*. Cover set C is then updated to $C = \{S_1, S_2\}$ where $S_2 = \{p_1, p_3, p_4\}$. The iteration is carried out for the rest of the GICSs and finally the cover set is $C = \{S_1, S_2, S_3, S_4, S_5\}$ where $S_1 = \{p_1, p_2\}$, $S_2 = \{p_1, p_3, p_4\}$, $S_3 = \{p_2, p_6\}$, $S_4 = \{p_6, p_7\}$, and $S_5 = \{p_5\}$.

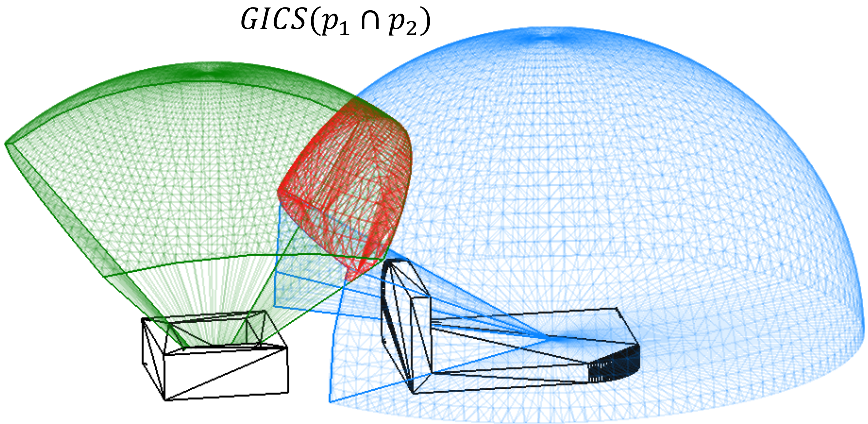
The process of intersecting GICSs is illustrated in Figure 7.5 (a)-(g). The shown instance comprises two objects and three inspection points measured by the same instrument with the same measurable space. The objective is finding the intersection among three GICSs that belong to the target points, respectively. p_1 is located on one surface of Object1 and p_2 is located in the pocket of Object2. Corresponding *GICS(p_1)* and *GICS(p_2)* are shown in Figure 7.5 (a) and (b) after visibility analysis. The Boolean operation takes place between *GICS(p_1)* and *GICS(p_2)* in Figure 7.5 (c) and the resulting intersection region $GICS(p_1 \cap p_2)$ is coloured in red in Figure 7.5(d). *GICS(p_3)* shown in Figure 7.5 (e) is a hemisphere originated at p_3 since there is no obstacle affecting its visibility apart from the surface it locates on. The same Boolean operation is conducted between $GICS(p_1 \cap p_2)$ and *GICS(p_3)* in Figure 7.5 (f) and the final $GICS(p_1 \cap p_2 \cap p_3)$ is obtained and presented using red triangulated mesh in Figure 7.5 (g). Under this circumstance, all three points are clustered in the same group and therefore can be measured by the same specified instrument located within this intersection region in single setup.



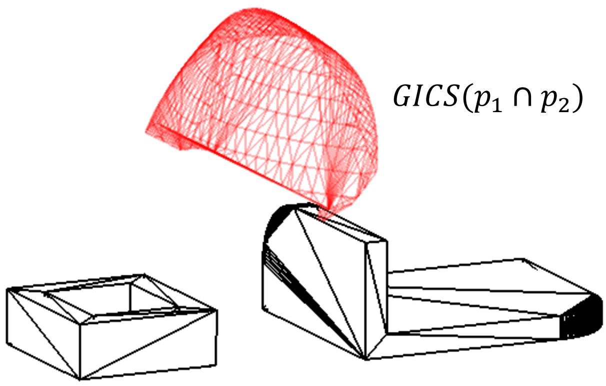
(a)



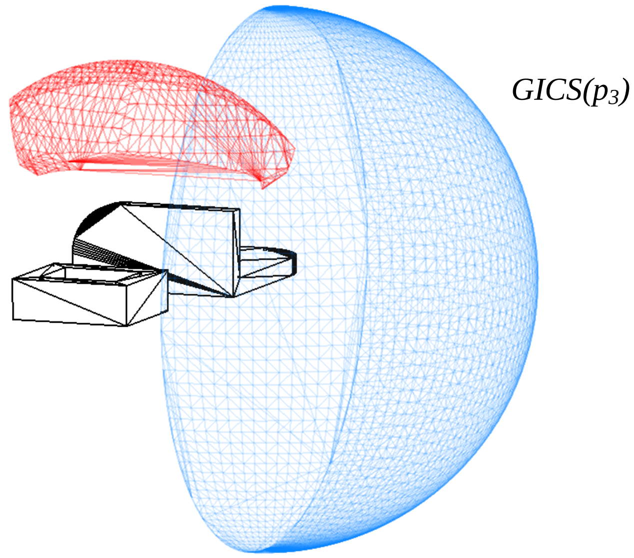
(b)



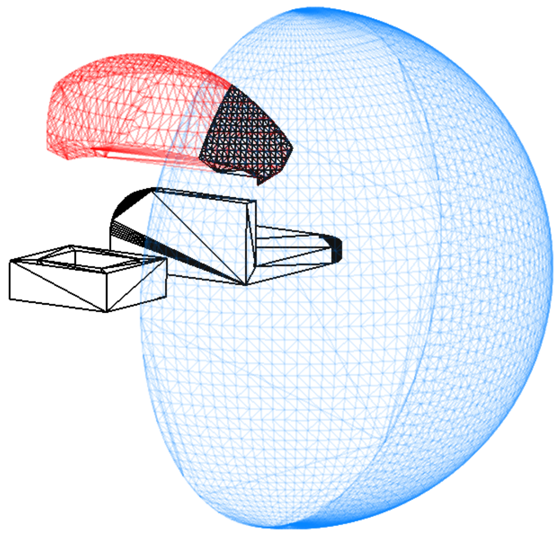
(c)



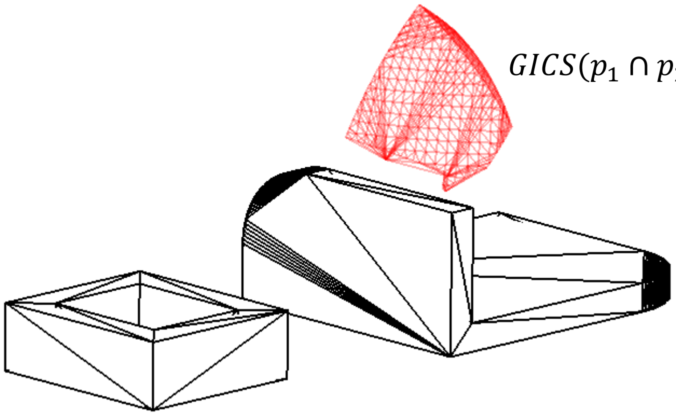
(d)



(e)



(f)



(g)

Figure 7.5 The process of intersecting GICSs

7.3 Summary

After the visibility analysis is carried out, it is mandatory to group and cluster the generated GICSs for all inspection points in order to find the minimum locations for LVM instruments. This can avoid unnecessary recalibration of the system and additional uncertainty caused by the location changes of the instrument. In the meantime, the efficiency of inspection is improved since the time of relocating the instrument is minimized.

Apart from the fact that no literature was found for clustering analysis for LVM applications, it is discovered that clustering techniques previously applied for CMMs and VMs are incapable of finding the appropriate cluster that cover all the 3D continuous configuration spaces. Set cover problem is innovatively selected to model the clustering process and a greedy heuristic is employed to find the approximately optimal solution. All inspection points are covered by the generated clusters while maximum coverage of each cluster is achieved.

CHAPTER 8 INSTRUMENT CONFIGURATION AND MEASUREMENT TASK SEQUENCING

8.1 Instrument configuration

8.1.1 Introduction

One characteristic of LVM is that the measured parts and products are comparatively large in volume. This leads to a clear difference compared with inspection on CMMs. Instead of fixing the part and orienting the measured target on a CMM, LVM instruments are mobilized around the part during inspection. Despite the clusters of GICSs being determined, the instrument still needs to be located at a certain position within related GICS. Since the visibility has been guaranteed, the objective function of this procedure is to minimize the total uncertainty to all target points. Nevertheless, the availability of uncertainty models for LVM instruments is severely limited at present. Therefore only the laser tracker is investigated in this study due to time limitation, but the proposed strategy can be applied to a variety of instruments if their uncertainty models are exposed.

8.1.2 Proposed methodology for laser tracker configuration

According to accuracy of the solution, two approaches are proposed, namely exact solution and discrete solution.

8.1.2.1 Exact solution

As far as laser tracker is concerned, the uncertainty of a measuring point located at *(X,Y,Z)* in 3D space is simply expressed as (Huo et al., 2010)

$$f(\boldsymbol{x}) = k(\boldsymbol{x}) + b \qquad \text{(Eq8.1)}$$

$$\boldsymbol{x} = (X - x_i, Y - y_i, Z - z_i) \qquad \text{(Eq8.2)}$$

Where k and b are specific coefficients with respect to different specifications of laser trackers and (x_i, y_i, z_i) are the 3D coordinates of the targets. The objective is to minimize *f(x)* constrained by $(X, Y, Z) \in GICS$ and determine the relative *(X,Y,Z)*. It can be expressed as:

$$min\, f(\boldsymbol{x}) = min \sum_{i=1}^{n} (k\sqrt{(X-x_i)^2 + (Y-y_i)^2 + (Z-z_i)^2}\ + b) \quad \text{(Eq8.3)}$$

This minimization process is an optimization of finding the minimum of constrained nonlinear multivariable function and can be solved efficiently using the optimization function, namely *fmincon()*, supported in Matlab Optimization Toolbox (MathWorks).

The *fmincon()* is able to find the minimum of a problem constrained by

$$\min_x f(x) \begin{cases} c(x) \leq 0 \\ c_{eq}(x) = 0 \\ \boldsymbol{A} \cdot \boldsymbol{x} \leq \boldsymbol{b} \\ \boldsymbol{A_{eq}} \cdot \boldsymbol{x} = \boldsymbol{b_{eq}} \\ \boldsymbol{l_b} \leq \boldsymbol{x} \leq \boldsymbol{u_b} \end{cases} \quad \text{(Eq8.4)}$$

where $\boldsymbol{x}$, $\boldsymbol{b}$, $\boldsymbol{b_{eq}}$, $\boldsymbol{l_b}$ and $\boldsymbol{u_b}$ are vectors, $\boldsymbol{A}$ and $\boldsymbol{A_{eq}}$ are matrices, c(x) and c_{eq}(x) are constraint functions that return vector outputs. Not only linear equations can be handled by $\boldsymbol{A_{eq}} \cdot \boldsymbol{x} = \boldsymbol{b_{eq}}$ and non-linear equation by $c_{eq}(x) = 0$, but also linear and non-linear inequality can be processed by $\boldsymbol{A} \cdot \boldsymbol{x} \leq \boldsymbol{b}$ and $c(x) \leq 0$, respectively. The upper and lower boundary of vector $\boldsymbol{x}$ is expressed by $\boldsymbol{l_b}$ and $\boldsymbol{u_b}$. The above available constraints pave the way of solving the 3D location selection problem within a spatial region as following. Eq8.5 is the objective function *f(x):*

$$fmincon(\boldsymbol{x}) = min \sum_{i=1}^{n} (k\sqrt{(X-x_i)^2 + (Y-y_i)^2 + (Z-z_i)^2}\ + b) \text{(Eq8.5)}$$

The constraints consist of different equations and inequalities depending on the spatial region obtained from the clustering analysis. Since CATIA is parametric CAD software, all surfaces and lines are generated by specific equations stored in the Geometric Modeller (DassaultSystèmes, 2000). By using the function CATSurface::GetEquation() supported in CAA, the boundary equations of the created entity representing the intersection of GICSs can be retrieved. This particular API is capable of computing the equations of surfaces as required and returning a constant pointer to the allocated memory where the equation is stored. By substituting the equations into Eq8.4 the constraints are determined and the objective function can then compute the optimized solution, which is the location of the point that has minimum uncertainty to all inspection points.

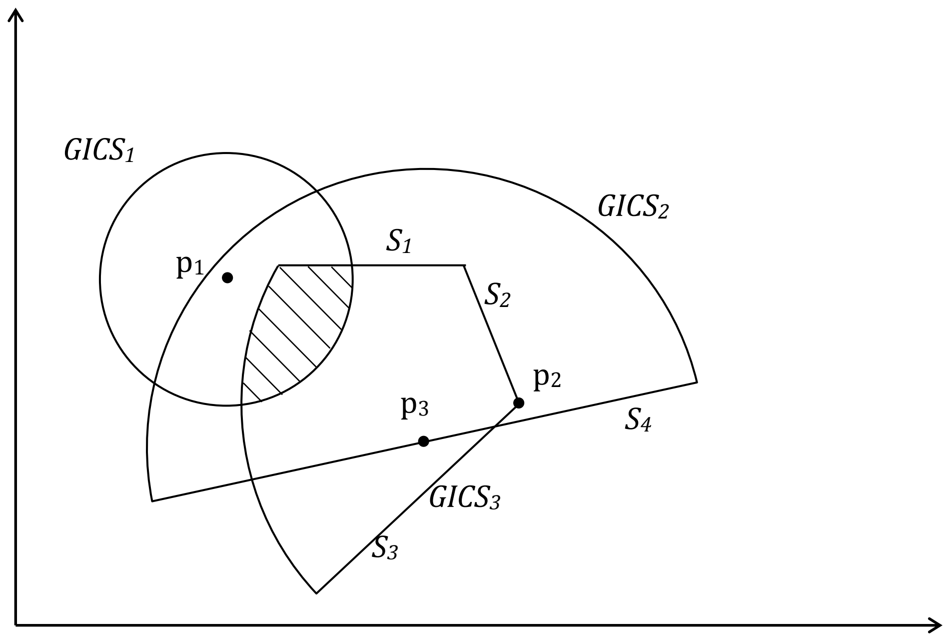


Figure 8.1 GICSs of three inspection points

Figure 8.1 is a demonstration of the proposed approach. There are three inspection points $p_1(x_1, y_1, z_1)$, $p_2(x_2, y_2, z_2)$, $p_3(x_3, y_3, z_3)$ with individual GICSs and the intersection of three GICSs has been computed such that all three points can be measured in the same cluster shown in the crossed area. The objective is to find the coordinates of a target point within the cluster, where minimum combined uncertainty with respect to p_1, p_2, and p_3 is obtained. The corresponding objective function is the combined uncertainty that can then be written as:

$$f(\boldsymbol{x}) = \sum_{i=1}^{3}(k\sqrt{(X-x_i)^2+(Y-y_i)^2+(Z-z_i)^2}\ +b) \qquad \text{(Eq8.6)}$$

Since the measurable spaces are all spheres, the primary spatial constraints are the three spherical regions, which can be expressed as:

$$\begin{cases} \sqrt{(X-x_1)^2+(Y-y_1)^2+(Z-z_1)^2} \le r_1 \\ \sqrt{(X-x_2)^2+(Y-y_2)^2+(Z-z_2)^2} \le r_2 \\ \sqrt{(X-x_3)^2+(Y-y_3)^2+(Z-z_3)^2} \le r_3 \end{cases} \quad \text{(Eq8.7)}$$

As non-linear inequalities, the constraint must be presented in the form of $c(x) \le 0$ and therefore the above constraints are converted to:

$$\begin{cases} \sqrt{(X-x_1)^2+(Y-y_1)^2+(Z-z_1)^2} - r_1 \le 0 \\ \sqrt{(X-x_2)^2+(Y-y_2)^2+(Z-z_2)^2} - r_2 \le 0 \\ \sqrt{(X-x_3)^2+(Y-y_3)^2+(Z-z_3)^2} - r_3 \le 0 \end{cases} \quad \text{(Eq8.8)}$$

Unlike $GICS_1$, $GICS_2$ is a hemisphere and $GICS_3$ is a part of a sphere, which results in the secondary constraints. The hemisphere can be considered the intersection of the sphere and the positive half space supported by the plane that has the normal vector as the pole, S_4 in this case. Likewise $GICS_3$ can be considered the intersection of the sphere, positive half space determined by S_3 and negative half space determined by S_1 and S_2.

A plane can be described using

$$ax + by + cz + d = 0 \quad \text{(Eq8.9)}$$

Let

$$D = \begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix} \quad \text{(Eq8.10)}$$

where (x_i, y_i, z_i) are the three points on the plane and by solving Eq8.9 the coefficients can be obtained:

$$a = \frac{-d}{D} \begin{vmatrix} 1 & y_1 & z_1 \\ 1 & y_2 & z_2 \\ 1 & y_3 & z_3 \end{vmatrix} \quad \text{(Eq8.11)}$$

$$b = \frac{-d}{D} \begin{vmatrix} x_1 & 1 & z_1 \\ x_2 & 1 & z_2 \\ x_3 & 1 & z_3 \end{vmatrix} \quad \text{(Eq8.12)}$$

$$c = \frac{-d}{D} \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \quad \text{(Eq8.13)}$$

d can then be solved by substituting the above equations back to Eq8.9.

After computing the equations of S_1, S_2, S_3 and S_4, the linear constraints in this instance can be subsequently expressed using the linear inequality $\boldsymbol{A} \cdot \boldsymbol{x} \leq \boldsymbol{b}$ as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \end{bmatrix} \leq \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \qquad \text{(Eq8.14)}$$

$$-\begin{bmatrix} x \\ y \\ z \end{bmatrix} \begin{bmatrix} a_3 & b_3 & c_3 \\ a_4 & b_4 & c_4 \end{bmatrix} \leq -\begin{bmatrix} d_3 \\ d_4 \end{bmatrix} \qquad \text{(Eq8.15)}$$

where [a_i, b_i, c_i, d_i] are the coefficients with respect to S_1, S_2, S_3 and S_4.

In this example there is no other constraint and therefore the objective function can be computed by means of calling the optimizing function *fmincon()*

$$x = fmincon(f(x), x0, \boldsymbol{A}, \boldsymbol{b}, \boldsymbol{A_{eq}}, \boldsymbol{b_{eq}}, \boldsymbol{l_b}, \boldsymbol{u_b}) \qquad \text{(Eq8.16)}$$

where $\boldsymbol{A_{eq}}$ =[], $\boldsymbol{b_{eq}}$ =[], $\boldsymbol{l_b}$ =[], and $\boldsymbol{u_b}$=[] and other constraints are shown in Eq8.8. The output of this optimization is the coordinates *(X, Y, Z)* of the target position where the minimum combined uncertainty is achieved.

8.1.2.2 Discrete solution

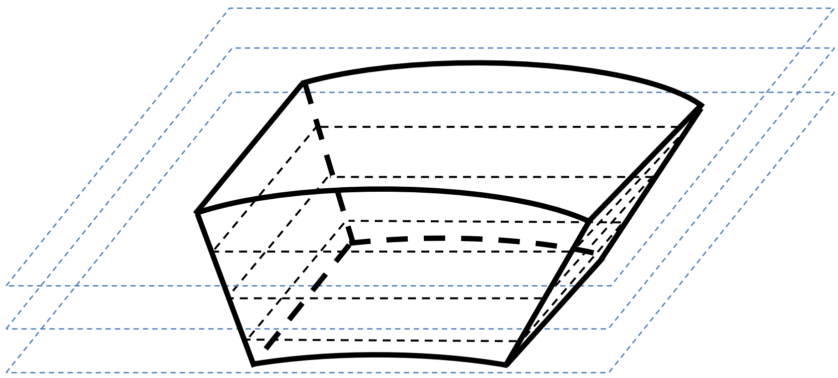
Although the previous method is able to provide the exact solution of the optimized target point, the application is limited by the availability of the uncertainty model for a specific instrument. In addition to that, the complexity of constructing the mathematical constraints as well as the acquisition of the surface equation brings in extra difficulties in practice. An approximate approach is proposed in this study employing the functionality supplied by the metrology software in order to cope with those deficiencies of the exact solution.

One characteristic of the clustered GICSs is that it is constrained within a certain spatial region. It is feasible to sample this region by a set of discrete points using predefined patterns that are subject to the boundary condition. The density of the point distribution is determined by the required resolution for instrument configuration. The measurement simulation function is then implemented at every point in the set outputting the combined uncertainties to all inspection points, which are stored accordingly in the uncertainty evaluation list. A simple search algorithm can query the list and identify the point with minimum combined uncertainty. The details of this approach are described below.

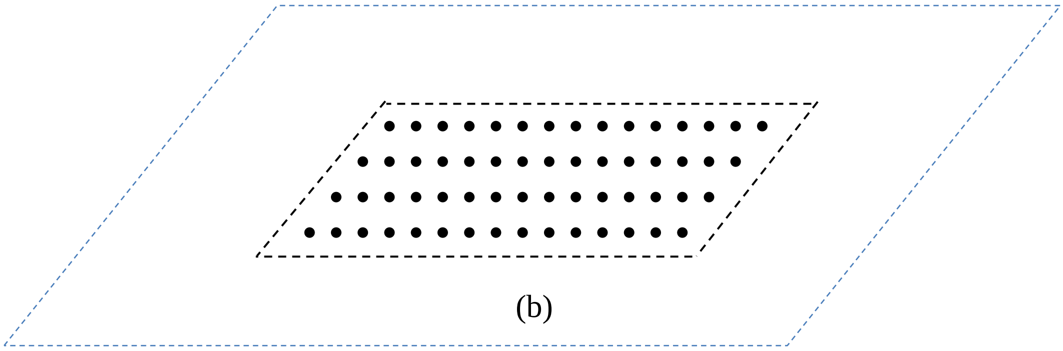
Step 1. Generate the set of discrete points representing the clustered GICS.

The density of the points set is primary concern in this step. As far as the LVM instruments are concerned, it is neither realistic nor efficient to select a resolution less than 10mm considering the entire volume of the working space and the process of locating the instrument. The points set can be created afterwards by intersecting a series of parallel planes, which parts from each other with the same resolution, to the clustered GICS. A point grid is then allocated to the intersecting section of each plane and the union of all point grids composes the discrete points within the GICS.

An illustrative example is shown in Figure 8.2 (a) and (b). The clustered GICS is the spatial region shown in bold lines and three parallel planes denoted by the blue dash lines are created to intersect with the GICS. The intersection planes are accordingly generated and the point grids can be located on each plane, shown in Figure 8.2 (b). Another example is given in Figure 8.3 (a) and (b) indicating the process in CAD environment. Six parallel planes are deployed to create the spatial points grid. For better illustrating purpose the points grid only on the bottom plane is visible.

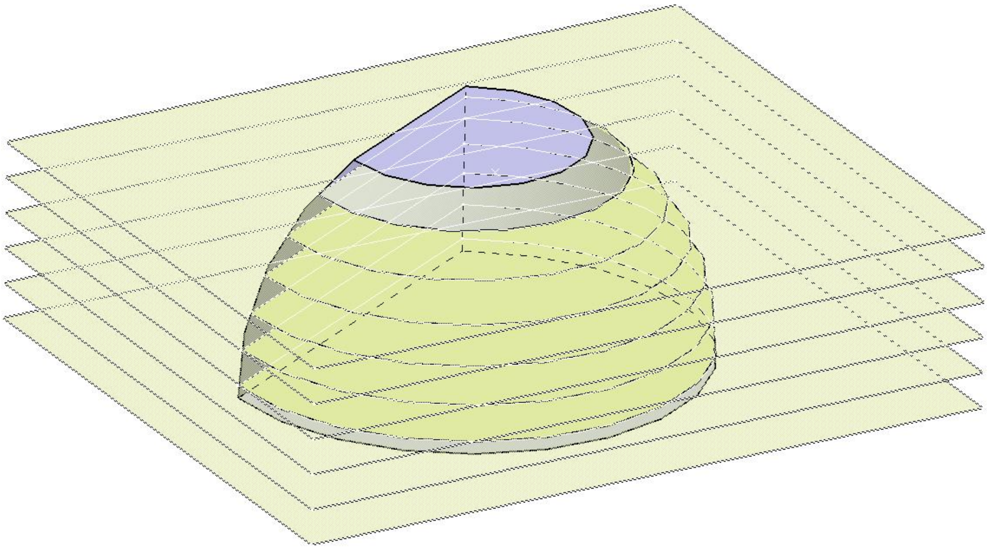


(a)

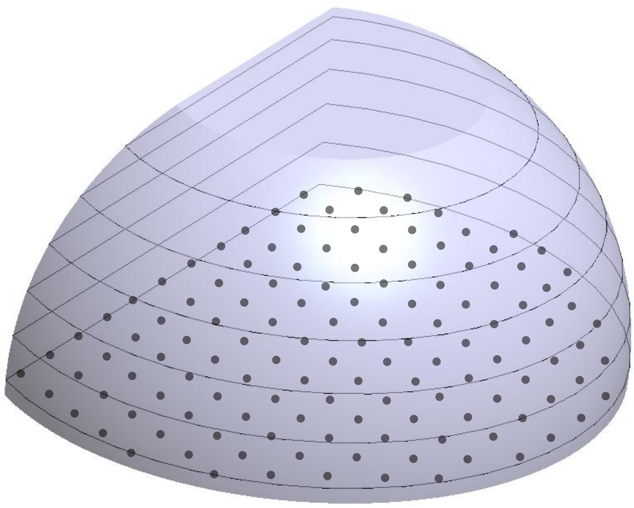


(b)

Figure 8.2 Discretization of the solution space



(a)



(b)

Figure 8.3 Discretization of the GICS using six planes

Step 2. Conduct simulated measurement and output the combined uncertainty.

Once all possible positions of the instrument are identified, the measurement simulation is conducted using specific instrument at those locations sequentially. The combined uncertainty to all inspection points is stored in a list shown in Table 8.1.

Table 8.1 Simulation of combined uncertainty to all inspection points

GICS ID	Instrument ID	Position	Combined Uncertainty
1	LT1	P1(112,35,49)	0.3544
1	LT1	P2(113, 38, 53)	0.3629
…	…	…	…

The combined uncertainty used here is the algebraic sum of individual uncertainties with respect to all inspection points. However, one may use the derived uncertainty of the objective inspection feature such as lines, planes, and circles. Figure 8.4 shows the screen shot of a measurement simulation process in SA that comprises two sets of points measured by a laser tracker. The simulated uncertainty of each point is computed by the software based on a predefined number of samples (Figure 8.5).

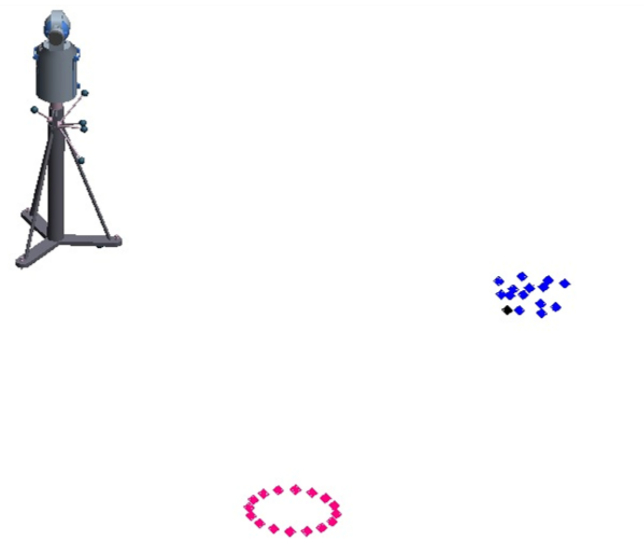


Figure 8.4 Simulation of the measurement uncertainty using SA

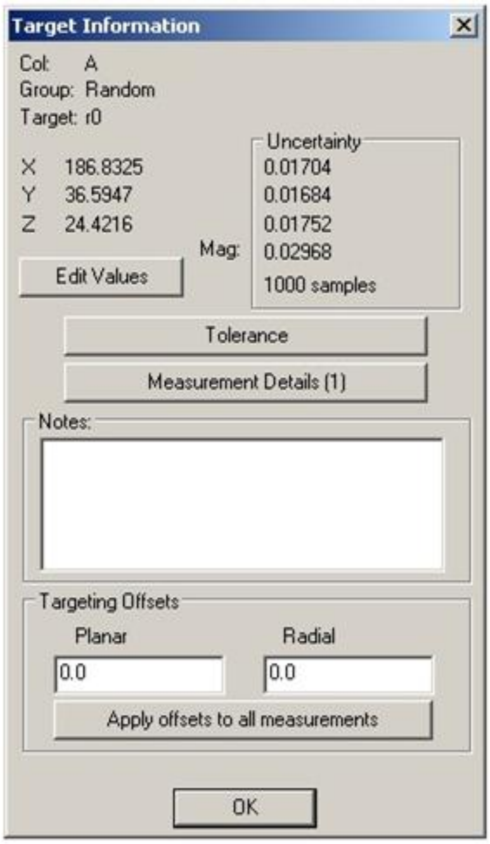


Figure 8.5 Simulated uncertainty information of an inspection point

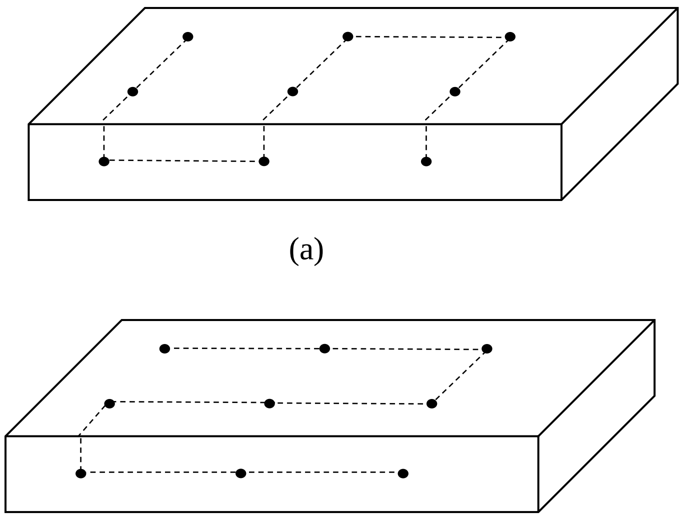
Step 3. Query for the point that yields the minimum combined uncertainty.

Having constructed the list that contains the uncertainty information with respect to all inspection points at every sampled location of the instrument, a simple search is able to return the ID of the location with the coordinates. Moreover, the query process can also identify the location that yields minimum uncertainty to a specific feature while the derived feature uncertainty is applied. This enables the process to manipulate the optimization result according to the priorities of different inspection features.

8.2 Measurement Sequencing

8.2.1 Introduction

Having gained the spatial configuration of instruments, the final assignment of inspection planning is to identify the measuring order of specified targets. It is evident that moving the optical sensor through the targets is one of the most time-consuming operations, especially while moving around discrete points on large-scale object.



(a)

(b)

Figure 8.6 Examine the part using different measuring sequence

Considering the simple example shown in Figure 8.6, nine inspection points are distributed on the surfaces of the cuboid and two inspection sequences are given in (a) and (b) respectively. It is surprisingly difficult to recommend either of the sequence as the optimal order using intuitionistic judgment in terms of the travelling distance along the path. Therefore, mathematical calculation is needed to compare the two distances. Assume the horizontal distance between every two adjacent points is x and the vertical distance between two adjacent points, including two points that are not on the same surface, is y. It is straightforward to derive the travelling distance of the path in (a) and (b) respectively:

$$T_1 = 6x + 2y \quad \text{(Eq8.17)}$$

$$T_2 = 6y + 2x \quad \text{(Eq8.18)}$$

It can be observed that the path in (a) is longer than (b) when $x > y$ and vice versa. Moreover, the difference between T_1 and T_2 is significant when the horizontal length and the vertical length differ considerably. It is therefore mandatory to take the sequence of the measurement into account in order to avoid the unnecessary travelling distance. This stage thus aims at minimizing the inspection time by identifying the shortest path among the targets.

8.2.2 Proposed path planning for LVM instruments

Although LVMIs share the same sampling strategy as the CMMs, specifically point measurements, there are essential dissimilarities existing between the two. CMMs can be considered a device operated by programmed robot such that the probe path is immutable once the program is written. Therefore the path must evade any potential collision to avoid damage to both the probe and the parts. On the contrary, LVMIs are primarily operated by the user except devices such as laser tracker and laser radar tracking the target automatically. The movement of the target is also determined and activated manually based on experience and intuition of the metrologist. This leads to a collision-free inspection but normally not an optimal path. Under this circumstance, it is realistic and practical to generate a guidance plan for LVMIs rather than an exactly executing program for sequencing the measuring points and planning inspection path.

The requirements of the sequencing and path planning for LVMIs are identified as follows:

1) The proposed approach must be applicable for all types of geometry regardless of the complexity.

2) All inspection tasks should be included in the planning process including the instruments involved at different locations.

3) An optimal or approximately optimal solution is expected to minimize the inspection time and cost.

In order to tackle the above issues, two levels of sequencing and path planning are required.

1) Sequencing the different locations of the employed instrument. This is the case that a specific instrument is located at different positions to complete the inspection. The order of those positions are determined first in a manner that the minimum travelling distance among the locations can be achieved. The sequence of locations where multiple instruments are placed is not constrained since the availability of instruments dominates the decision and simultaneous inspection is allowed. Note that multiple instruments indicates the number of the device regardless the type. For instance, two laser trackers are considered multiple instruments since they can be operated at the same time.

2) Sequencing and planning the path for the inspection points within the same cluster. Having sequenced different positions of the specific instrument, the micro level of planning aims to provide the operator a guidance to measure all the targets within every cluster respectively. The generated sequence and path is the optimal or approximately optimal trajectory in terms of minimizing the travelling distance. The starting and ending points can be assigned by the user or the planner chooses the points automatically to maximize the efficiency.

The problem of finding the shortest path among a set of points with constraints is deemed as the traveling salesman problem. By solving the TSP using existing approaches, optimal solution or approximate optimal solution can be obtained. Therefore the sequencing and path planning for LVMIs is modelled as a 3D TSP in this study and the details are given in the following sections.

8.2.2.1 Introduction to TSP

The TSP was researched initially in 18th century by a number of mathematicians and included in the content of graph theory (Biggs et al., 1986) The definition can be stated as (Biggs et al., 1986):

‘Given a set of cities and the cost of travel (or distance) between each possible pairs, the TSP, is to find the best possible way of visiting all the cities and returning to the starting point that minimize the travel cost (or travel distance).’

It can be expressed using the following mathematical mode:

Given a set of cities $C = \{c_1, \cdots, c_n\}$, the edge length between each pair of cities$l(c_i, c_j)$, the objective is to find a minimal length closed tour so that every city must be visited only once.

The problem is symmetric TSP (sTSP) when $l(c_i, c_j) = l(c_j, c_i)$ and becomes asymmetric (aTSP) when there is at least one pair that $l(c_i, c_j) \neq l(c_j, c_i)$. An important extension of the traditional TSP is the multi travelling salesman problem (mTSP). A number of salesmen are traversing all the cities in such a manner that each city is visited exactly once while the total distance is minimal. Despite the simple concept behind the problem, it is proved NP-hard since the number of solution reaches $(n-1)!/2$ for aTSP where n is the number of the cities.

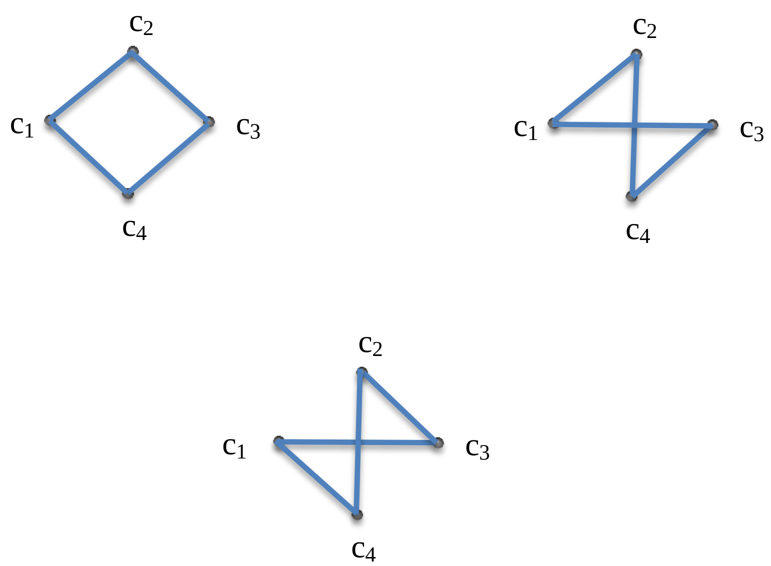


Figure 8.7 An aTSP example with all solutions

An example of aTSP consisted of 4 cities is shown in Figure 8.7. Three tours are subsequently obtained traversing all cities yet each city is only visited once. The distances of three tours are

$$d_1 = |c_1c_2| + |c_2c_3| + |c_3c_4| + |c_4c_1| \quad \text{(Eq8.19)}$$

$$d_2 = |c_1c_2| + |c_2c_4| + |c_4c_3| + |c_3c_1| \quad \text{(Eq8.20)}$$

$$d_3 = |c_1c_3| + |c_3c_2| + |c_2c_4| + |c_4c_1| \quad \text{(Eq8.21)}$$

The optimal tour can be determined by comparing the length of each tour using algebraic sum. However, this method can only be applied when the number of city is relatively small due to the enormous searching space. There are 181,440 tours if 10 cities are considered and $3.04140932 \times 10^{62}$ tours if 50 cities are involved, which is neither realistic nor applicable using brutal force to compute the solution.

Apart from the significance in computational theory, the TSP has been applied and linked with many applications including crew scheduling (Svestka and Huckfeldt, 1973), vehicle routing (Lenstra and Rinnooy Kan, 1974), computer wiring (Lenstra and Rinnooy Kan, 1975), circuit boards drilling (Grötschel et al., 1991), mission planning (Brummit and Stentz, 1996; 1998). One can conclude that the TSP is a typical model of problems involving resource planning for multiple tasks. The mathematical model was successfully applied to several inspection planners for CMMs as well.

Many researchers have strived to solve the TSP using a variety of formulations and the integer programming formulation appears to be the most applied mathematical formulation, indicated by the (Orman and Williams, 2006; O¨ncan et al., 2009). In particular, Applegate et al. (2003; 2006) proposed a binary formulation to describe the most effective exact algorithm. In their study, every distance between two cities was assigned with a binary value that was equal to one if and only if this specific path was in to the optimal solution. The objective was to minimize

$$\sum_{i<j} l_{ij} x_{ij} \quad \text{(Eq8.22)}$$

subject to

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad \text{(Eq8.23)}$$

$$\sum_{i,j\epsilon S} x_{ij} \le |S| - 1, (3 \le |S| \le n - 3) \qquad \text{(Eq8.24)}$$

$$x_{ij} = 0 \; or \; 1 \qquad \text{(Eq8.25)}$$

Eq8.23 is the degree constraint that ensured every city appeared in the optimal tour only once by confining that the salesman departure from the city i for one time and enter the city j for one time only. Eq8.24 eliminates any subtour in the solution by removing the possibility of any loop and Eq8.25 maintains the integrity of the formulation.

Solving this integer problem is extremely laborious due to the enormous solution searching space. In fact, the problem was tackled when first proposed by Dantzig et al. (1954) at a size of only 49 cities and after 55 years an algorithm named Concord developed by Applegate et al. (2009) was able to provide the solution to the TSP involving 85,900 cities with a cost of 136 CPU years.

As the exact solution is enormously expensive in terms of modelling and computational cost, approximate algorithm or heuristic algorithm is widely used to solve the TSP in real applications. The approximate algorithms often generate a solution within certain degree of deviation from the optimal solution whilst only a feasible and affordable computational resource is needed. Due to the huge desirability, a number of approximate algorithms were developed and applied such as closest neighbour heuristic, greedy heuristic, insertion heuristic, Christofide heuristic, tabu search algorithm, simulated annealing algorithm, genetic algorithm, ant colony optimization, artificial neural networks, fuzzy algorithm, particle swarm optimization etc. (Laporte, 1992; Cho, 2010; Lu and Xie, 2010; Matai et al., 2010; Mo, 2010). The performance and the polynomial complexity of some of the above algorithms are given in Table 8.2.

8.2.2.2 TSP modelling of the LVMIs measurement sequencing

1) Modelling of the TSP

As the outputs from previous inspection planning stages are a number of clusters including all inspection assignments, two levels of sequencing and path planning are required:

(1) Sequencing the different locations of the employed instrument.

(2) Sequencing and planning the path for the inspection points within the same cluster.

Table 8.2 The performance and the polynomial complexity of some algorithm

Algorithm	Complexity	Performance (deviation from the optimal solution)
Closest neighbour heuristic	$O(n^2)$	25%
Greedy heuristic	$O(n^2 \log_2(n))$	15%-20%
Insertion heuristic	$O(n^2)$	Not available
Christofide heuristic	$O(n^3)$	10%
Tabu search	$O(n^3)$	Less than 5%
Simulated annealing	$O(n^2)$	Less than 5%
Genetic algorithm	$O(n^2)$	Less than 5%

In order to make the genetic algorithm applicable, the above levels can be modelled as two TSPs.

2) TSP for sequencing the instrument locations

This is the higher level of sequencing and path planning aiming to obtain a minimum traversing distance for a specific instrument among all the determined locations.

Considering the instance shown in Figure 8.8, the specific inspection assignments require to locate the laser tracker in a number of positions around the part. Assume that there is only one instrument and the locations are spatially distributed in the working volume to measure all the surfaces of the part, which is placed above the ground although the fixtures and gigs are not shown here. The objective is to identify the movement of the laser tracker with minimum traversing distance.

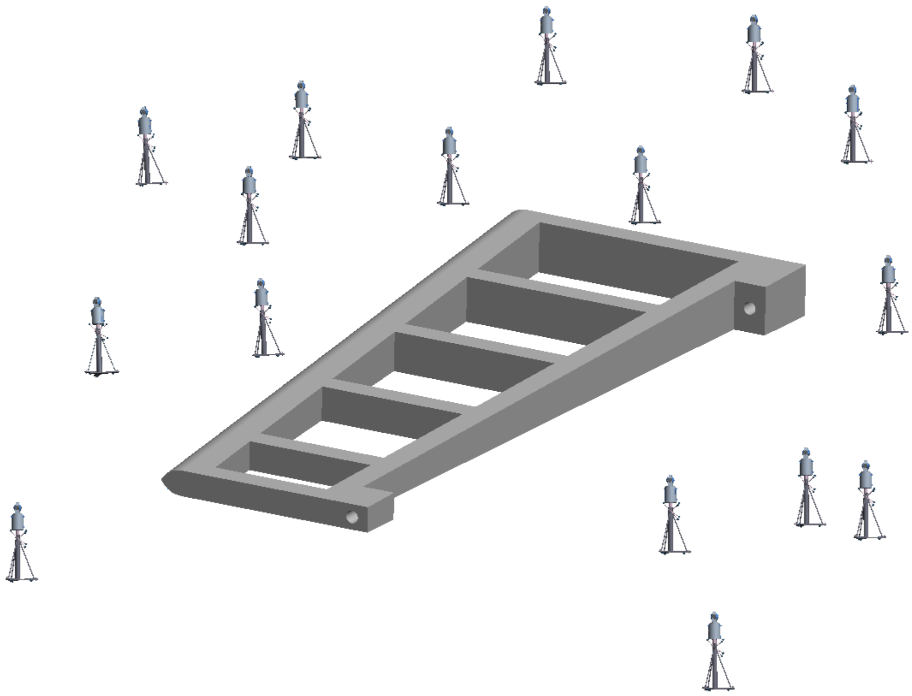


Figure 8.8 Example of modelling inspection locations to TSP cities

Two requirements must be satisfied in the sequencing process:

1) The planned path must not cross through the open space in the part since it is not safe and realistic to move the instrument with the risk of colliding the part. Therefore the instrument must be relocated around the part.

2) The starting location and end location can be fixed or open depending on the particular case. The algorithm must be able to provide flexible solution.

Under this circumstance, it can be abstracted as a 3D TSP with constraints in solution space shown in Figure 8.9. Each instrument location is described by a spatial point in the coordinate system with individual coordinates. The distance between every pair of cities is expressed by the Euclidean distance. The inspection part is bounded using a cube to avoid any path crossing through the part. The bounding cube can be obtained by setting a safety distance from the part, which guarantees that there is no collision between the path and the part if the path is not penetrating the cube.

The modelled TSP can then be solved by genetic algorithm explained later in this chapter with two types of generated path: collision-free path and path penetrating the bounding cube. The latter path implies that there is no existing route that connects the corresponding cities without colliding the safety volume. It is subsequently treated using the following approach, shown in Figure 8.10, to avoid the interference.

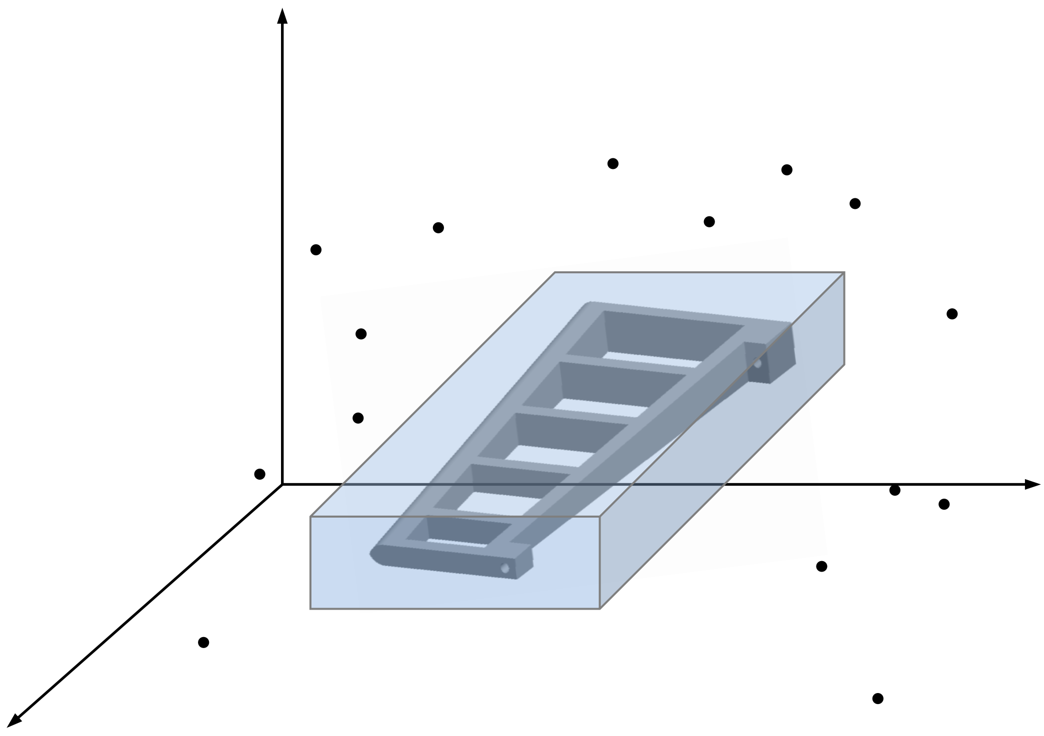


Figure 8.9 Modelled TSP with safe bounding volume

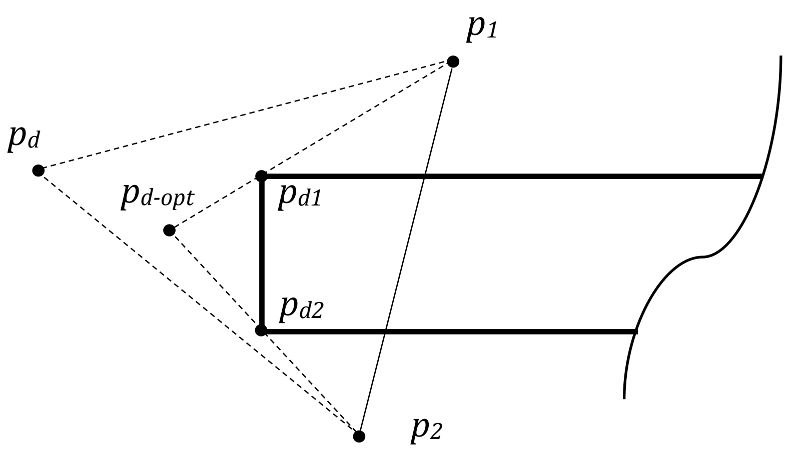


Figure 8.10 The proposed approach to modify penetrating path

One traditional method coping with the situation that $\overline{p_1p_2}$ penetrates the safety volume is to create a dummy point p_d. Path $\overline{p_1p_d}$ and path $\overline{p_dp_2}$ then replace the original route to eliminate the collision. The inserting of dummy points is a manual process according to the observation of the operator and the quality of the position varies. In the example the optimal position is at p_{d-opt}. A new approach is proposed here to automate the process with a better-outputted path. Two dummy points p_{d1} and p_{d2} are inserted instead of one by projecting the original path $\overline{p_1p_2}$ to the bounding cube. The new path is consequently obtained, which is consisted of $\overline{p_1p_{d1}}$, $\overline{p_{d1}p_{d2}}$, and $\overline{p_{d2}p_2}$. This path is better than the re-routing method using a single dummy point and can be proved as follows:

Assume the p_{d-opt} can be found by the operator and the entire distance is:

$$l_1 = \overline{p_1p_{d-opt}} + \overline{p_{d-opt}p_2}$$

$$= \overline{p_1p_{d1}} + \overline{p_{d1}p_{d-opt}} + \overline{p_{d-opt}p_{d2}} + \overline{p_{d2}p_2} \quad \text{(Eq8.26)}$$

The two-points method leads to a path with the distance:

$$l_2 = \overline{p_1p_{d1}} + \overline{p_{d1}p_{d2}} + \overline{p_{d2}p_2} \quad \text{(Eq8.27)}$$

Considering the triangle$\Delta p_{d-opt}p_{d1}p_{d2}$, according to the law of cosines:

$$\overline{p_{d1}p_{d-opt}} + \overline{p_{d-opt}p_{d2}} > \overline{p_{d1}p_{d2}} \quad \text{(Eq8.28)}$$

Comparing Eq8.26 and Eq8.27, it can be concluded that

$$l_2 < l_1 \quad \text{(Eq8.29)}$$

3) TSP for sequencing the inspection points

This is the micro level of sequencing and path planning for all inspection points within one cluster that can be measured by the instrument at one location. Each point is modelled as one city of the TSP and all the cities must be visited only once to complete the inspection. However, the path between a pair of cities is expressed using different methods depending on the different types of employed instruments.

1) Photogrammetry system. Since the instrument can measure a group of points simultaneously at a certain direction, the sequences and paths among the points are ignored. The system is rotated to capture all the points. An illustrative example is shown in Figure 8.11.

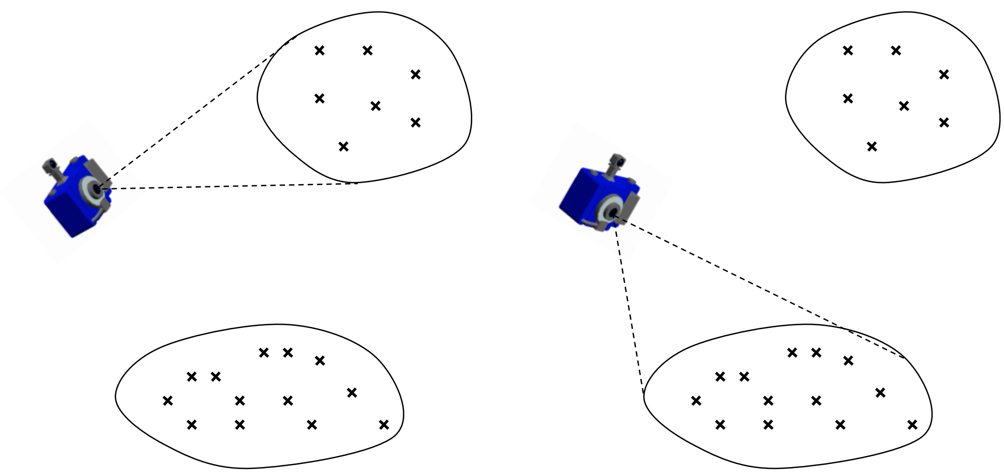


Figure 8.11 Inspection two sets of points by changing the direction of the photogrammetry system

2) Instrument without targeting device. For instruments such as laser radar, total station, theodolite and laser scanner, the measurements are carried out by emitting and receiving the signal. Due to the absence of the physical targeting device, the movement of the projecting signal is not affected and constrained by the geometry of the part and surroundings after the visibility of each target has been guaranteed previously. The distance of the route between a pair of cities is therefore modelled by 3D Euclidean distance. The path shown in

Figure 8.12 represents the trajectory of the signal beam although it penetrates the surface at the edge fillet twice.

3) Instrument with targeting device. For instruments such as laser tracker, iGPS and laser arm, contact with the inspection point is mandatory since either a sensor or probe must be placed at the measuring target. The distance of the route between a pair of cities is determined based on the locations of the cities, which can be classified as follows.

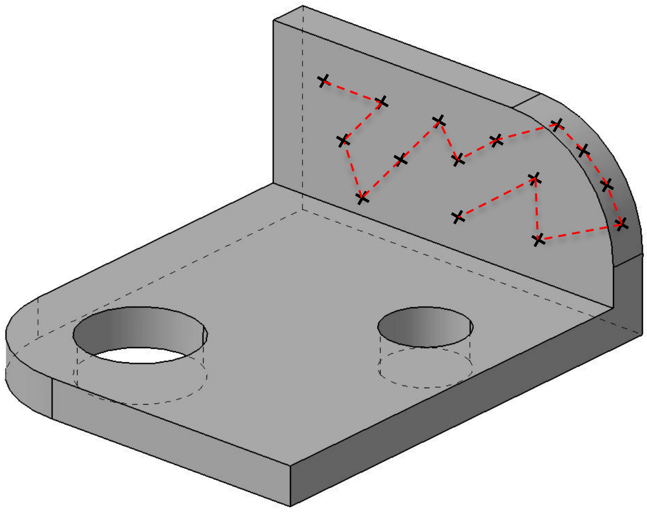


Figure 8.12 Path planning without consideration of obstacles

a. Cities are located on planar surface.

   3D Euclidean distance is calculated to represent the distance of the path between the cities. The targeting device can be manoeuvred along the path without interference with the surface.

b. Cities are located on surfaces that form a concave region.

   3D Euclidean distance is calculated to represent the distance of the path between the cities. The targeting device can be manoeuvred along the path without interference with the surfaces.

c. Cities are located on surfaces that form a convex region.

   Geodesic shortest distance is calculated to describe the distance of the path in order to avoid the penetrating path through the surfaces. The targeting device can be moved along the generated geodesic shortest path on surfaces. The computation of geodesic paths and distances on polyhedrons is the problem of finding the shortest distances between points on surfaces in 3D volume. It has been studied and applied to a variety of applications in areas including robotics, geographic information systems, circuit design, and computer graphics (Kanai and Suzuki, 2001; Maheshwari and Wuhrer, 2009). A number of exact and approximate algorithms are available at present (Sharir and Schorr, 1986; Mitchell et al., 1987; Chen and Han, 1996; Kimmel and Sethian, 1998; Martinez et al., 2004; Surazhsky et al., 2005; Aleksandrov et al., 2006; Xin and Wang, 2007) to compute the exact geodesic path between two points on a polyhedron object. The details of related algorithms are beyond the scope of this research and the reader is recommended to refer to the literature for comprehensive information.

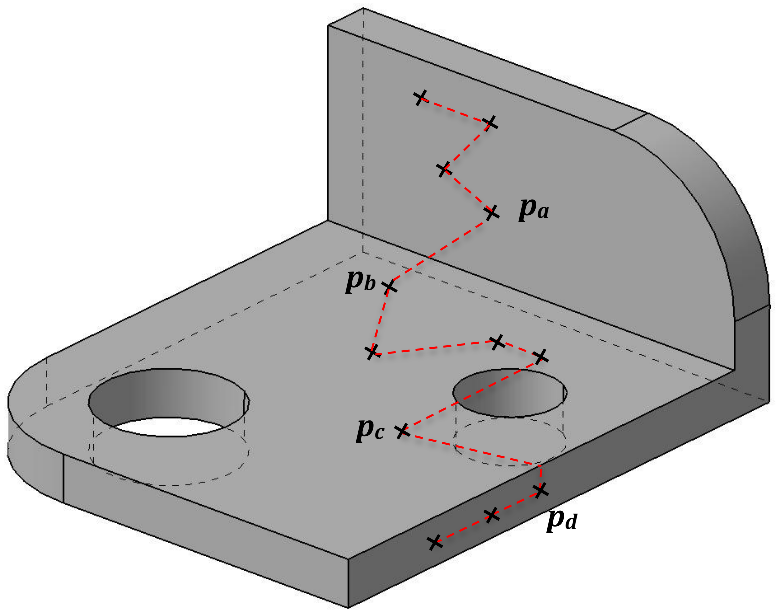


Figure 8.13 Path planning using 3D Euclidean distance and geodesic distance

Considering the example shown in Figure 8.13, paths between points apart from $\overline{p_a p_b}$ and $\overline{p_b p_c}$ are all planar routes decided by the 3D Euclidean distance. Points p_a and p_b are in concave region therefore the Euclidean distance is applicable whilst p_b and p_c are in convex surfaces where the geodesic path must be produced to avoid the interference between the targeting device and the part.

8.2.2.3 Genetic algorithm for the modelled TSP

Genetic algorithm (GA) has been widely applied as a search heuristic for several decades in fields such as computational science, engineering, mathematics and physics (Davis, 1985; Kolen and Pesch, 1994; Lu et al., 1996; Potvin, 1996; Lu et al., 1999; Cheng et al., 2002). By borrowing the nature evolution concepts such as inheritance, mutation, selection and crossover, GA is able to provide approximate solutions to many optimization and search problems (Goldberg, 1989) and Lawler et al. (1985) successfully applied GA to solve the TSP in 1985. A variety of new approaches based on GA were developed for TSP following the first success to improve the performance and obtain more accurate results such as immune-genetic algorithm (Zeng and Gu, 2007; Qi et al., 2008; Itoh, 2010; Lu and Xie, 2010), hybrid genetic algorithm (Nguyen et al., 2007) and multi-world intelligent genetic algorithm (Onoyama et al., 2000; Sakurai and Tsuruta, 2010). It can be concluded that GA is one of the best approximate algorithms for TSP due to its robustness, efficiency and affordable computational cost with a 2%-3% deviation from the mathematical optimal solution. Therefore, this study adopts the GA to obtain the solution for inspection sequencing and path planning.

1) Introduction to GA

The initial setup in a genetic algorithm is to construct a population of strings, namely chromosomes, representing possible solutions to the problem. Evolution is carried out at each generation from a randomly generated population using different genetic operators including crossover, inversion and mutation. In each generation, a fitness evaluation process takes place for existing candidates and acceptable solutions are maintained to form the next generation with a number of genetically evolved individuals. Better solutions are expected progressively until the predefined number of generations has been iterated or the objective fitness level is achieved. A near optimal solution can be obtained if the number of generations is set to a reasonable value. The primary processes of a typical genetic algorithm can be expressed using Figure 8.14.

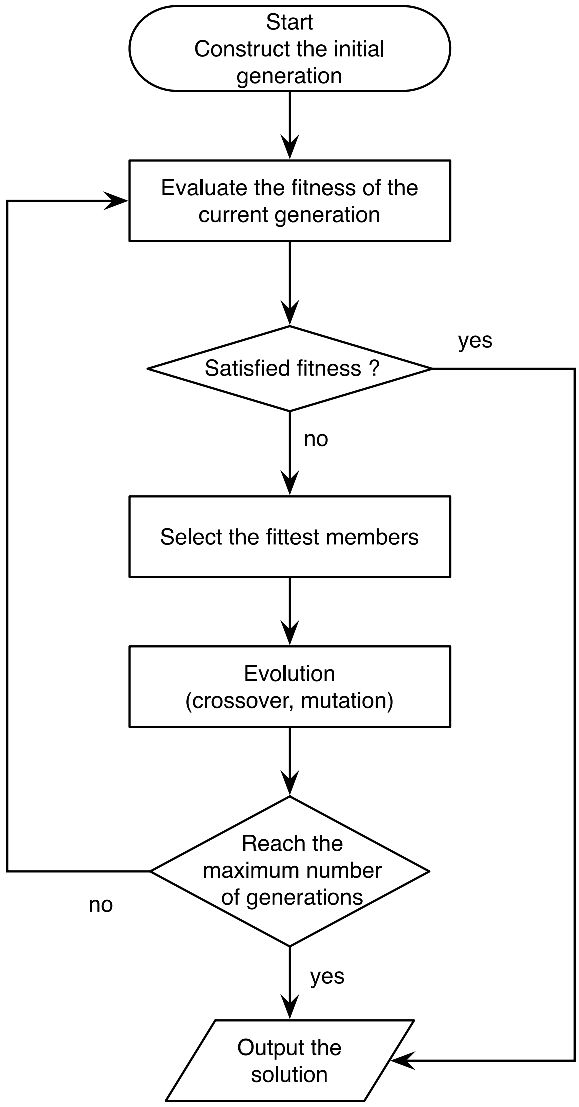


Figure 8.14 Typical processes of a genetic algorithm

2) The proposed genetic algorithm

As described in the previous section, two levels of TSPs are obtained after modelling the specific sequencing and path planning process. A genetic algorithm is subsequently developed for solving this optimization problem. The overview of the proposed algorithm is shown in Figure 8.15. An initial population of candidates is created followed by evaluating the fitness of each individual. A selection process takes place according to their performance in the evaluation and the surviving candidates are recombined using genetic operators such as crossover and mutation to produce new evaluated offspring. The iteration of this evolution process is conducted until the desired number of generation is achieved. Details of the algorithm are given in the following sections.

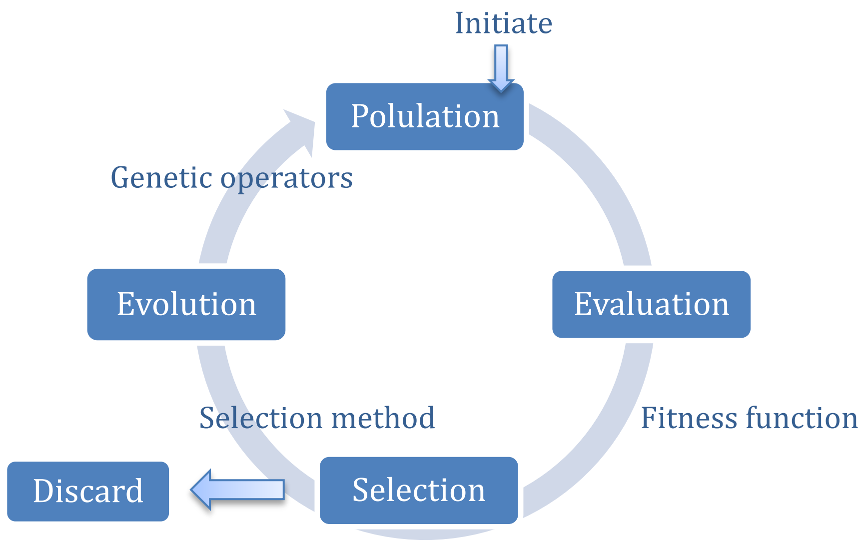


Figure 8.15 Proposed genetic algorithm

Encoding:

The initial step of developing a GA is to choose an encoding scheme for creating the chromosomes. There are three well-applied encoding schemes, namely binary encoding, value encoding and permutation encoding. As the first working encoding scheme with GA, binary encoding stores the gene information using binary numbers which is effective for many applications. Nevertheless, it has been revealed that the disadvantages are significant due to the large number of cities in TSP, resulting an enormous size of array (Lu et al., 1999; Nguyen et al., 2007; Itoh, 2010; Sakurai and Tsuruta, 2010). Value encoding is normally adopted when the problem involves complex values such as real numbers, characters and phrases (Obitko, 1998; Boukreev, 2001). Permutation encoding is widely applied for ordering problems such as TSP due to its inherent ability of indicating the order among the chromosome. In this study, aiming to generate an optimized tour of all inspection points, the chromosome should be constructed as a string of all the targets. Permutation encoding is chosen and conducted such that the sequence of the gene within the chromosome indicates the ordering of inspection points.

The typical structure of constructed chromosomes is shown in Figure 8.16. Every inspection point is assigned with an ordinal when the TSP is modelled. Each chromosome contains the entire set of cities and the visiting sequence is determined by the locus in the array. Therefore the length of the chromosome is the number of cities and every ordinal must be present only once in the array. Chromosome A and B are two solutions of a TSP consisted of 9 cities and the visiting sequence is 2-3-5-8-1-4-6-7-9 and 8-1-6-5-3-9-4-7-2, respectively.

Chromosome A	2	3	5	8	1	4	6	7	9
Chromosome B	8	1	6	5	3	9	4	7	2

Figure 8.16 Structure of chromosomes

Initialize the population:

After encoding all cities into chromosomes, the initial population must be generated, namely the initial tour of all cities. The tour can be generated randomly, inputted by a human expert, or provided by a heuristic method (Al-Dulaimi and Ali, 2008; Sakurai and Tsuruta, 2010). Researchers have pointed out that a good initial population results in shorter converging time (Cheng et al., 2002; Zeng and Gu, 2007; Itoh, 2010). A greedy searching algorithm is applied here: starting from a random point, the algorithm always choses the nearest point as the next destination of the path. If it is an inspection process with a fixed starting point, the algorithm is carried out from this point to construct the initial population. The complexity of greedy heuristic is $O(n^2 \log_2(n))$ thus it can generate the tour very quickly (Matai et al., 2010). However, since a greedy algorithm is constantly selecting the nearest city as the next destination, it is possible that the nearest city has appeared in the tour. In this case, another random city is selected as the next travelling position, resulting in a number of initial tours. An operator-inputted value, specifically the initial population size, defines the number of tours generated at the beginning for subsequent selection and evolution. It has been revealed that a large size of initial population cost more to obtain a solution whilst smaller size leads to the possibility of missing the optimal solution (Obitko, 1998; Boukreev, 2001).

Evaluation:

The performance of each chromosome in the population must be evaluated so that the following evolution process can be conducted on chromosomes with relatively weak score. The quality of the tour in this problem is the length of the traversing path. Therefore the fitness function is constructed as:

$$Fitness\_chromosome = \sum_{i=1}^{n-1} Distance(p_i p_{i+1}) \qquad \text{(Eq8.30)}$$

where *i* is the locus of the chromosome and *n* is the number of the cities.

The distance calculation in the fitness function is carried out based on the spatial relation between the pair of cities, which has been analysed when the TSP was modelled previously. 3D Euclidean distance and the shortest geodesic distance are employed to compute the distance of the path and the fitness of each solution is accordingly obtained afterwards.

Selection method:

The best candidates should survive and generate evolved offspring, which is the key theory in Darwin' evolution theory (LaLena, 2010). This allows the children inherit the excellent genetic information from their parents, which accomplishes the evolution. There are several approaches available to GA to select the best chromosomes such as roulette wheel selection, Boltzman selection, tournament selection, rank selection, and steady state selection (Al-Dulaimi and Ali, 2008; Itoh, 2010). Most proposed GAs for TSP used the roulette wheel selection due to its effectiveness and therefore it is adopted in this study. A comparative study on the performance of different selection methods can be found (Boukreev, 2001).

Implied by its name, the roulette wheel selection method chose the candidates based on the randomness of a roulette wheel. Instead of assigning equal probability to every number in a real game, a better-performed chromosome is accompanied with a wider "slot" on the wheel, which leads to greater probability of being selected by the wheel. The selection process is simulated as:

Step1. The sum of all fitness values in the population is computed.

$$Sum\_Fitness = \sum_{k=1}^{m} Fitness\_chromosome_k \quad \text{(Eq8.31)}$$

where k is the ordinal number of the chromosome and m is the population.

Step2. A random number R is generated in the interval *(0,Sum_Fitness).*

$$R = random(0, Sum\_Fitness) \quad \text{(Eq8.32)}$$

Step3. The fitness value of each chromosome in the population is summed up in succession until the sum is equal or greater than R. The current chromosome is then selected as the surviving parent.

An example of roulette wheel selection is illustrated in Figure 8.17. Chromosome 1 and 2 are more likely to be selected in Step3 as their fitness values constitute most part of the *Sum_Fittness*, which are indicated by the percentages of chromosomes.

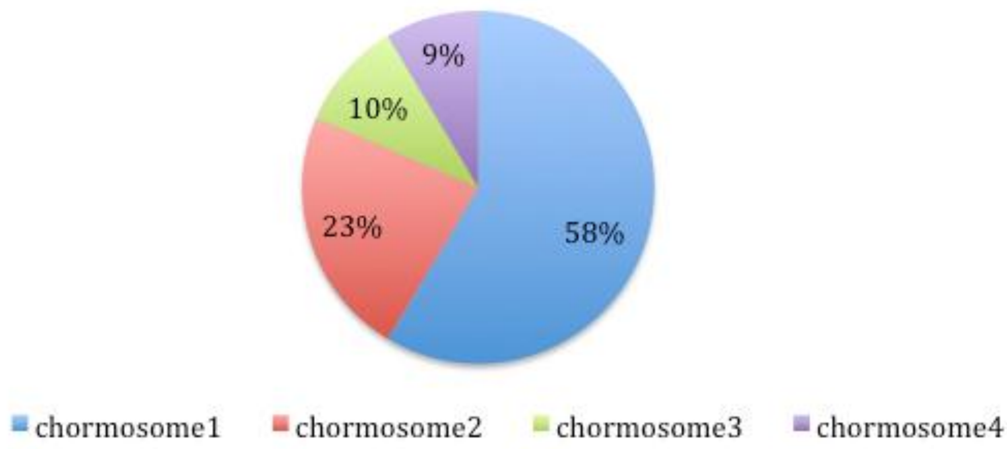


Figure 8.17 Roulette wheel selection of chromosomes

Evolution:

The evolution of chromosomes is the core section of a genetic algorithm, which ensures the progressive improvement of the solution. A new generation is produced using genetic operators on selected parents. Elitism, crossover and mutation are applied in the proposed approach.

1) Elitism

One of the main drawbacks of GA is the inadvertent loss of potential optimal solutions due to the evolution on random chromosomes. Elitism improves this issue by duplicating the best solution with the highest fitness in the present population into the new generation.

2) Crossover

Crossover is the main genetic operator to generate new offspring. Traditionally, two selected parent chromosomes are divided into two sections at the same position and the offspring is consisted by one section from each parent. The process is shown in Figure 8.18.

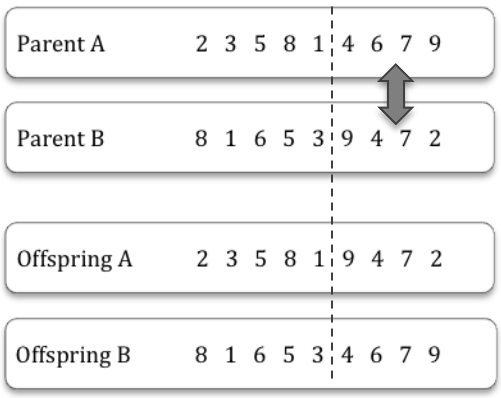


Figure 8.18 Crossover process

It can be noticed that the generated offspring are not validated since the same city appears more than once in one tour. A special crossover method, specifically Oder Crossover (OX), is developed to overcome this issue (Buckles et al., 1990). Take the chromosomes shown in Figure 8.18 as an example, two cutting positions are first randomly selected and the genes within the segment are copied to the offspring at the same position as follows:

Chromosome A	2	3	5	|	8	1	4	6	|	7	9
Chromosome B	8	1	6	|	5	3	9	4	|	7	2

Offspring A	-	-	-	|	8	1	4	6	|	-	-
Offspring B	-	-	-	|	5	3	9	4	|	-	-

Chromosome B is then used to create the rest genes of offspring A. The starting gene in the child A is copied from the second cut point of chromosome B, followed by the other genes:

Offspring A*	7	2	8	1	6	5	3	9	4
Offspring B*	7	9	2	3	5	8	1	4	6

The temporary offspring are then compared with previous generated offspring A and B to eliminate repeated genes:

Offspring A*	2	-	-	-	5	3	9	-
Offspring B*	-	2	-	-	8	1	-	6

The remaining genes are then placed into offspring A and B and the final results are:

Offspring A	7	2	5	|	8	1	4	6	|	3	9
Offspring B	7	2	8	|	5	3	9	4	|	1	6

3) Mutation

Mutation is another vital method controlling the evolution. The genetic algorithm is known to suffer from local optimization during the crossover process. Mutation at a certain probability is capable of decreasing the chance of algorithm converging due to local optimal solution. Again, the tradition mutating approach of changing the gene randomly cannot be used to obtain a valid tour after the mutation. A swapping operation that exchanges the positions of two randomly selected genes is applied here to maintain the validity of the new tour. It can prevent local optimization effectively although more computational cost is required (Onoyama et al., 2000;

LaLena, 2010; Sakurai and Tsuruta, 2010). For instance, by swapping the third gene with the sixth gene in the given chromosome A, a new chromosome B is generated:

Chromosome A	2	3	5	8	1	4	6	7	9
Chromosome B	2	3	4	8	1	5	6	7	9

8.2.3 Implementation of the genetic algorithm and experimental results

The workflow of the proposed genetic algorithm is shown in Figure 8.19. The algorithm is coded within Matlab. The input of the algorithm includes:

(1) *matrix _positions*: the $(n \times 3)$ matrix of points denoted by the 3D coordinates;

(2) *matrix_distance:* the $(n \times n)$ distance matrix;

(3) *pop_size:* the initial population size;

(3) *iteration:* the maximum number of generations.

The output of the algorithm is the approximate optimal solution obtained within the assigned maximum number of generations, accompanied by the total cost of the solution as well as the computation time.

Two examples are given here to demonstrate the effectiveness of the proposed genetic algorithm solving the inspection sequencing and path planning. The results also verify the modelling process of TSP is success and effective. The tests are carried out on a computer with i7-2.3G quad core CPU and 4G memory installed. Although the computer is capable of processing data using eight simulated cores, the code was written in single process programming and therefore a higher clock rate is more beneficial than multi-core in terms of computational time.

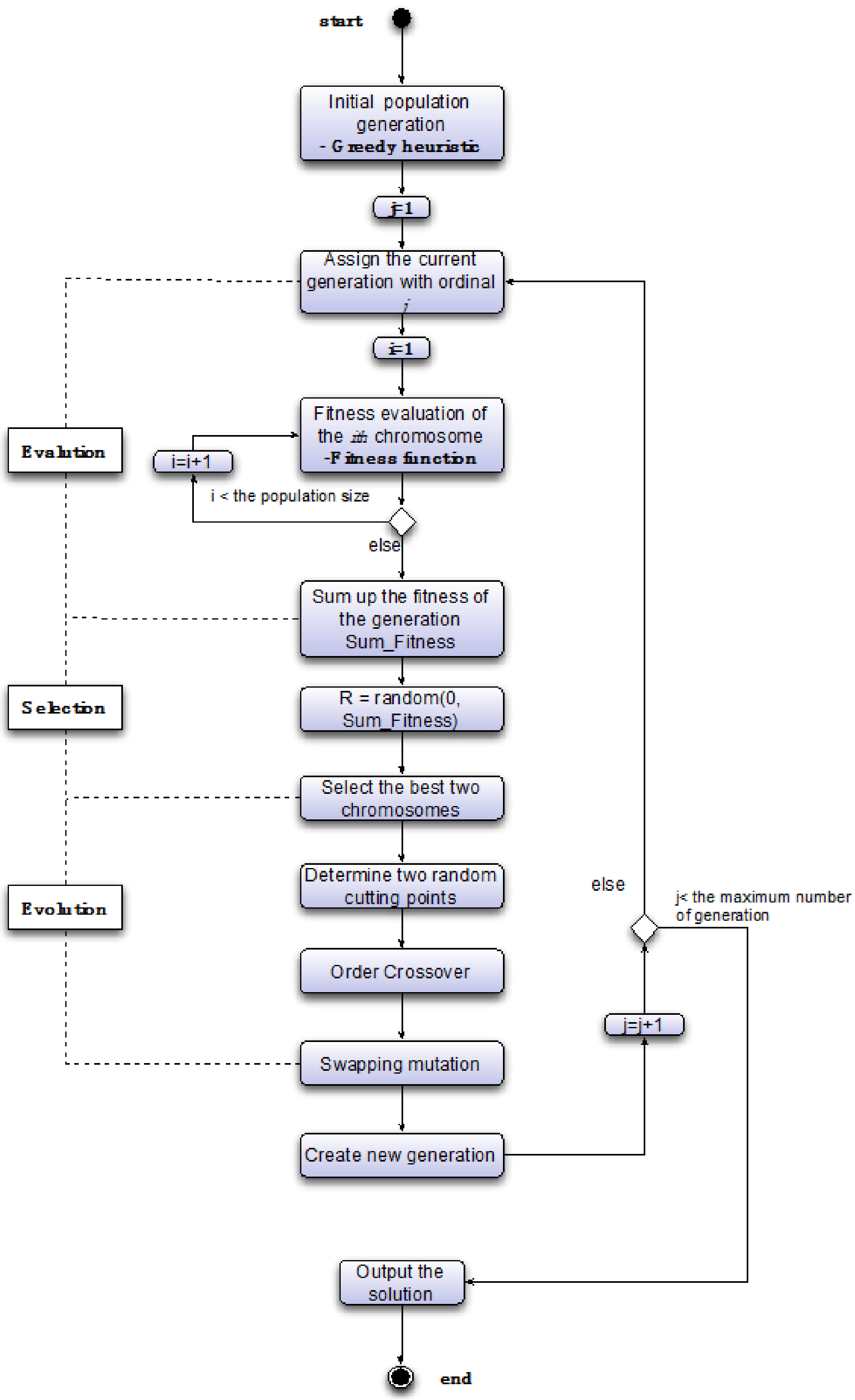


Figure 8.19 The proposed genetic algorithm

The first example is the higher level of sequencing and path planning, which determines the order of instrument locations and the guidance path traversing all the positions. The given inspection task employs one instrument and 30 instrument locations have been identified from the previous visibility and clustering analysis. The objective is to generate a sequence and path plan for the operator to manoeuvre the instrument in an optimal manner. No interference between the path and the inspection part is considered in this instance. The spatial distribution of the locations is given in Figure 8.20 and the matrix of the 3D coordinates of all points is inputted into the algorithm as initial parameters. A series of tests are carried out with different initial population size and the maximum number of generations. Table 8.3 shows the running results of the algorithm at different initial setups.

Table 8.3 Running results of the instrument location planning

pop_size	Iteration	Computation time(s)	Path length (units)
30	10^3	0.86	162.1307
	10^4	1.69	161.6429
50	10^3	1.03	159.6264
	10^4	2.49	158.3155
100	10^3	3.44	158.3155
	10^4	6.90	158.3155

The results clearly indicate the same observation with (Boukreev, 2001; Itoh, 2010; Sakurai and Tsuruta, 2010), A small size of initial population takes less time to converge but the optimal solution is likely to be missed. More generations can slightly improve the result with additional computational cost but optimal solution is still unable to achieve since the algorithm is confined with local optimization. With the increasing size of population size, better paths can be obtained and the optimal solution can be found at the 7831th iteration with a *pop_size* of 50. The same path can be retrieved with at the 860th iteration with a *pop_size* of 100 and no better path can be found with more iterations.

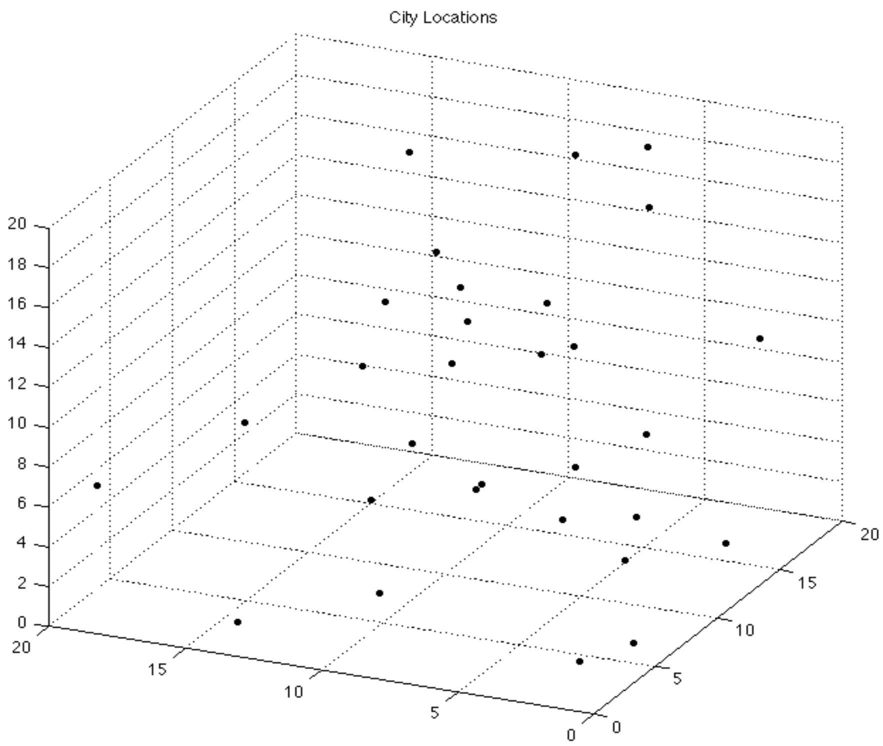


Figure 8.20 Instrument locations

Figure 8.21 shows the sequencing result with a starting population size of 30 and 1000 iteration while Figure 8.22 illustrates the optimal path among all the locations with initial population size of 100 and 1000 generations. The paths differ at the lower part of the point distribution, leading to the difference in the length of two tours. The best solution with the *pop_size* of 100 is presented in Figure 8.23 with *x* axis showing the iteration and *y* axis denoting the length of the solution. The convergence of the computation occurred at 860th generation, which was 3.23s after the initiation of the algorithm.

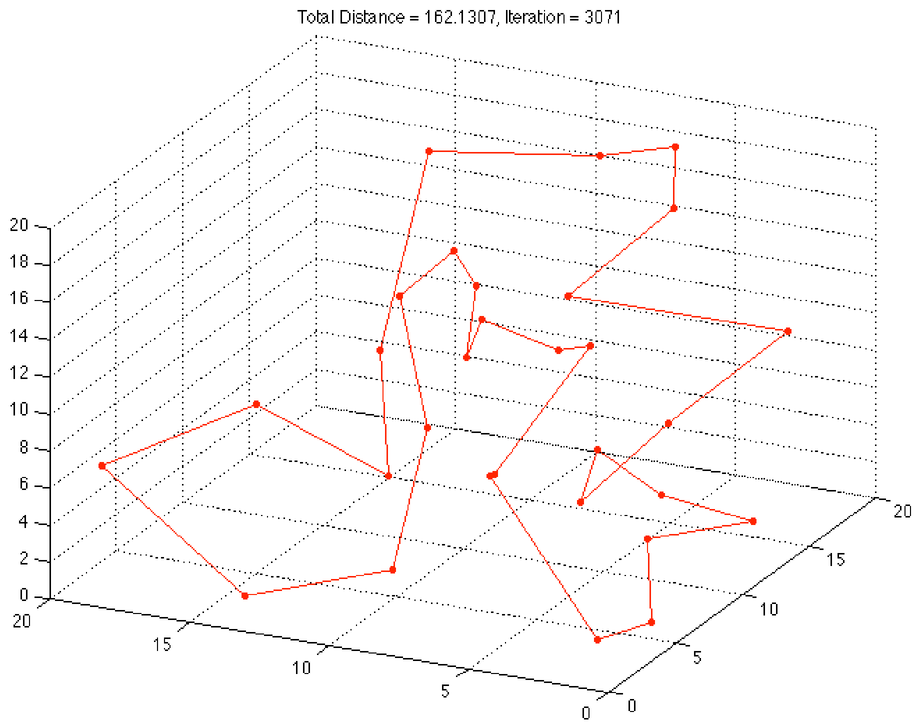


Figure 8.21 Sequencing result with pop_size of 30 and 1000 iteration

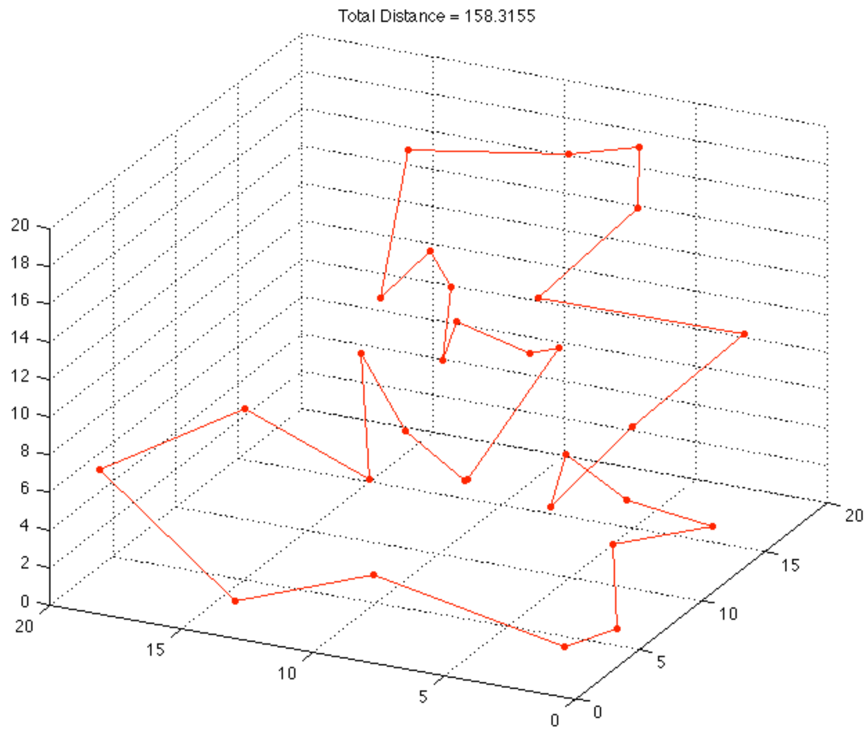


Figure 8.22 Sequencing result with pop_size of 100 and 1000 iteration

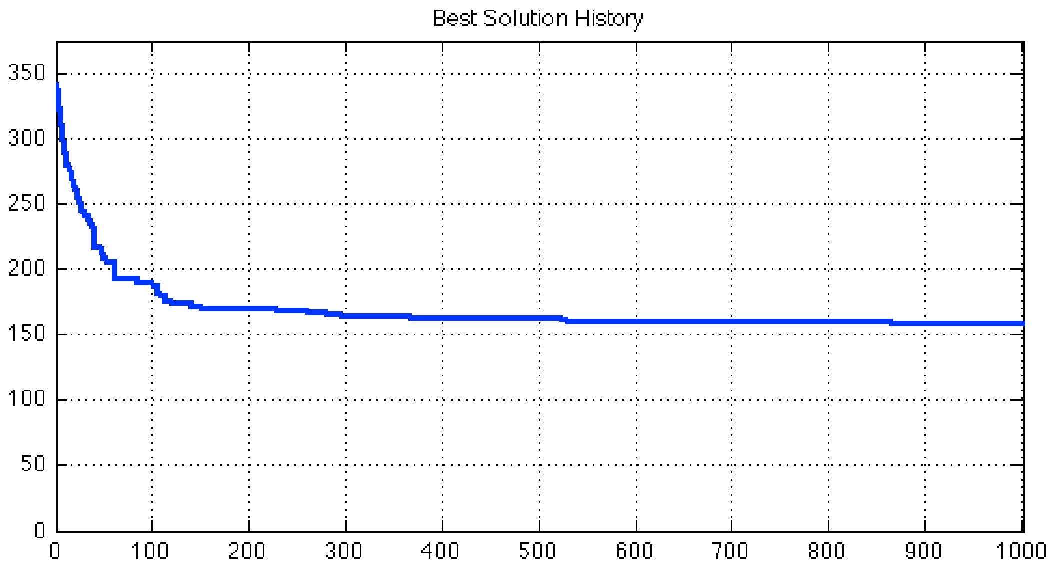


Figure 8.23 The solution history of the optimal sequence planning

The second example is the lower level of sequencing and path planning, where the inspections points are modelled as cities in TSP and the genetic algorithm is applied to find the optimal tour among all the points. Figure 8.24 shows an example of inspection part with 245 allocated target points. Each point is modelled as a city that a salesperson has to visit only once. The optical sensor travels throughout those points and there exists one optimal tour that has minimum total distance yet without sub-tour. The measuring results are utilized to compute the actual deviation of the surface from CAD model. Shown in Figure 8.25, the spatial distribution of inspection points is inputted into the genetic algorithm and 3D Euclidean distance is used to present the distance between each pair of points. Since the targeting device is not required to return to the starting point, the algorithm is slightly modified to eliminate the closed-path constraint. Table 8.4 reveals the results of the algorithm with different initial parameters.

Table 8.4 Running results of the measurement sequencing process

pop_size	Iteration	Computation time(s)	Path length (units)
60	10^4	15.33	96.7358
	2×10^4	23.18	82.7731
	4×10^4	56.71	80.1074
80	10^4	18.49	94.3106
	2×10^4	38.25	79.3249
	4×10^4	74.36	79.2373
100	10^4	24.31	92.4220
	2×10^4	55.49	83.5738
	4×10^4	92.79	76.6375
	6×10^4	130.28	74.1641
	8×10^4	269.21	74.3271
120	10^4	32.08	91.0254
	2×10^4	68.23	79.6934
	4×10^4	101.39	76.3559

The behaviour of the genetic algorithm in this example is distinct with the previous instance. Although the length of the tour is gradually shorter with the increasing value of *pop_size,* the quality of the tour is relatively poor at low iteration even when the initial population includes 120 tours. On the contrast, the length of the tour is decreased significantly with increasing number of generations in the range from 10^4 to 6×10^4. It can be seen that the result is not improved after the number of iteration exceeds a certain value, where worse path occurs. Moreover, the computational cost increases enormously with the number of iteration whilst the improvement of the tour is only on a small magnitude. Throughout the experiments, the best tour with a path length of 74.1641 units was achieved when the setup contained the initial population of 100 at 58864th iteration. The generated 3D path with length 79.3249 units is shown in Figure 8.26(a) and Figure 8.26(b) presents the overlook of the path. Figure 8.27 is the obtained best path and the difference between the two tours is significant. The best solution history is presented in Figure 8.28 with *x* axis showing the iteration and *y* axis denoting the length of the solution.

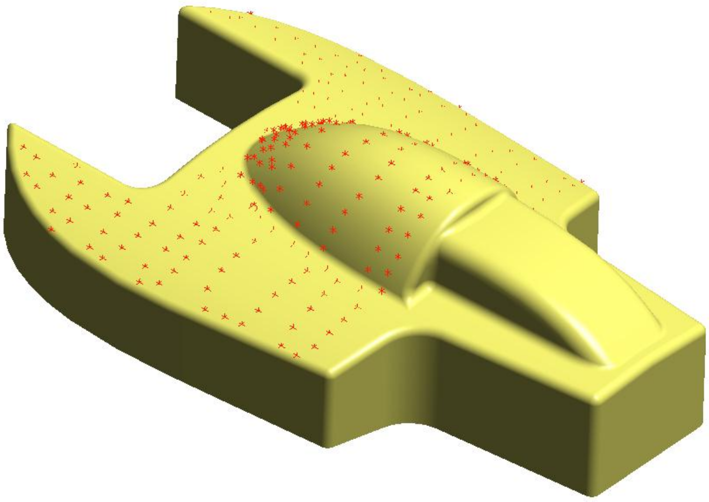


Figure 8.24 Inspection points on the part

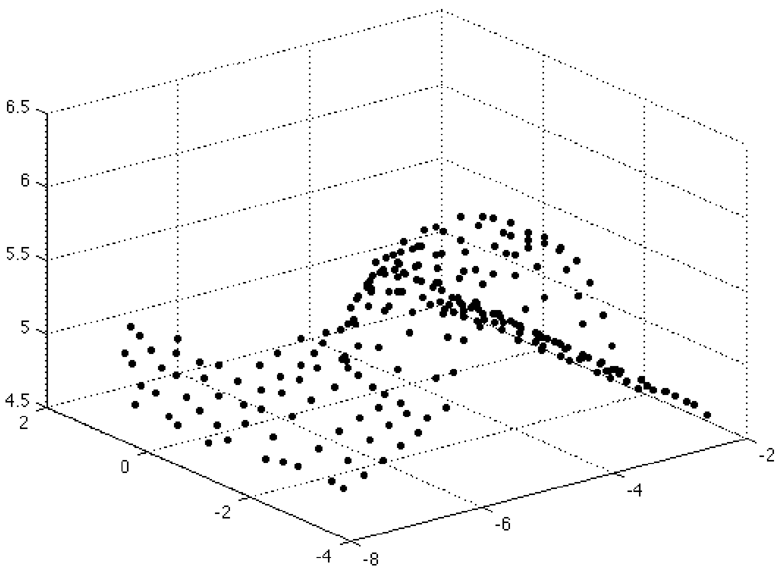
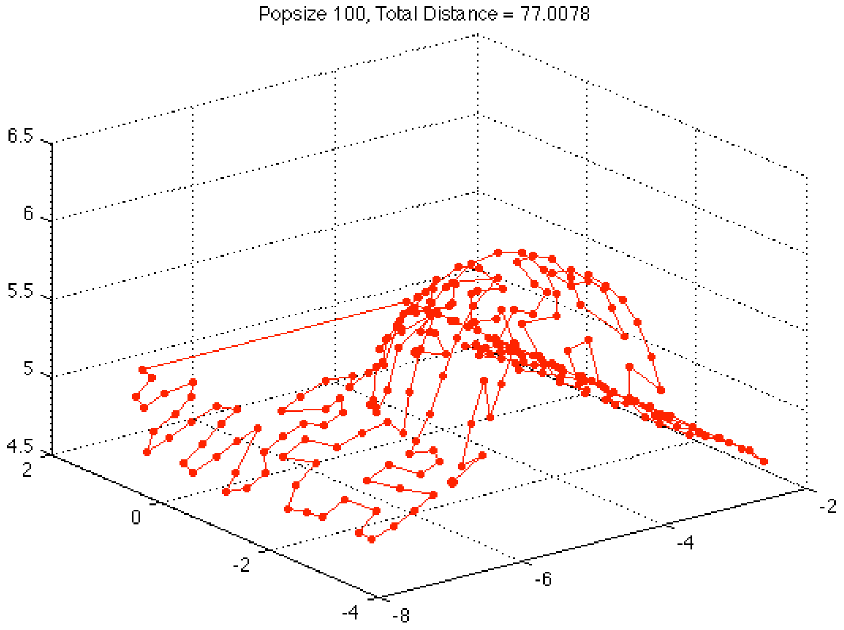
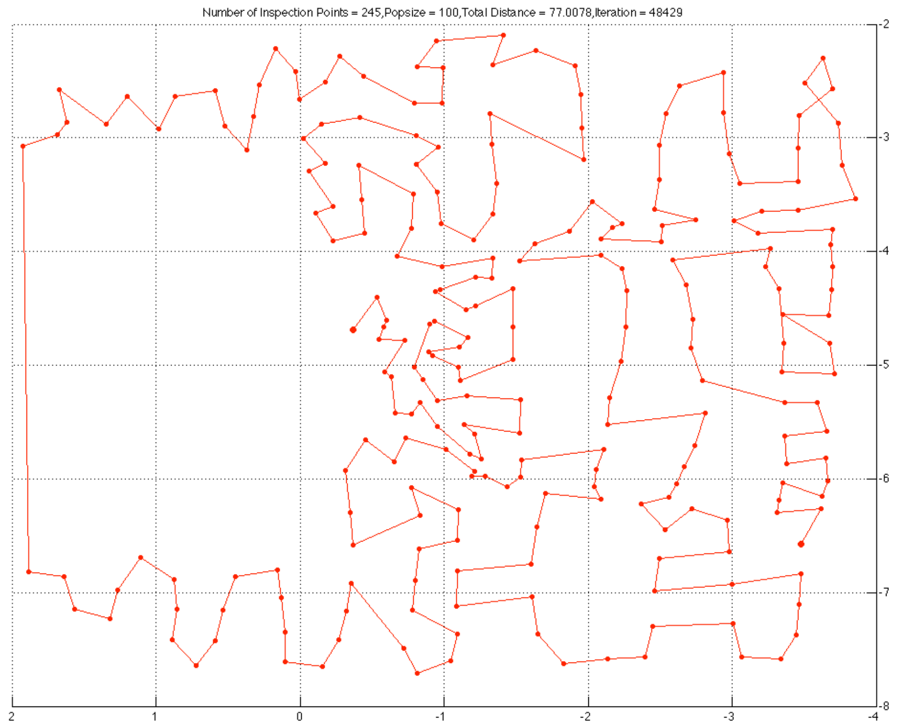


Figure 8.25 Imported locations of the inspection points



(a)



(b)

Figure 8.26 Generated 3D path for the inspection process

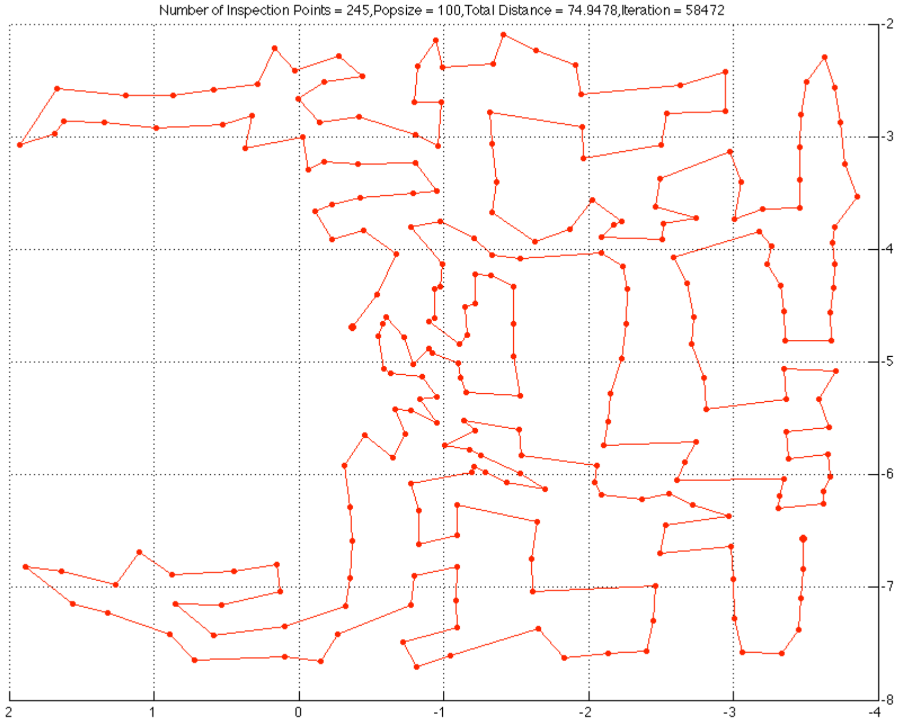


Figure 8.27 Approximate optimal inspection path

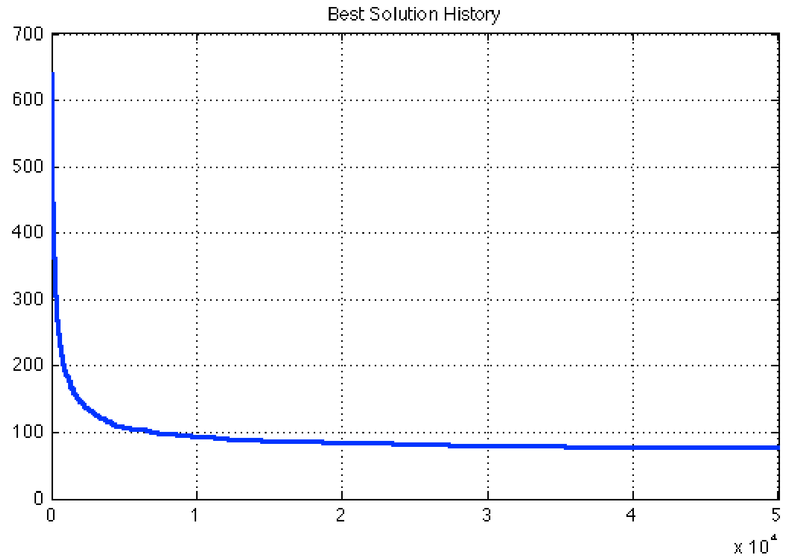


Figure 8.28 Solution history of the approximate optimal path

8.2.4 Discussion

Apart from the two examples given above, a number of numeric tests were conducted. A few observations and suggestions are given as follows for achieving an acceptable result with affordable computational resource.

The higher level of sequencing and path planning normally contains locations less than 30. The corresponding TSP can be modelled and solved by the proposed genetic algorithm efficiently in less than 5 seconds. In order to obtain an optimal solution, the recommended initial population size is 100 and the maximum number of generation should be 10^4. A verification of the generated path can be conducted using a different setup and the same configuration of the tour is expected if it is the optimal solution.

The sequencing and path planning for inspection points is unlikely to generate the optimal path if the number of targets is considerably large, specifically larger than 50. This is owing to the possibility of existing tours increasing factorially. As this process aims to provide the operator guidance for manoeuvring the targeting device, the path is not exactly as the actual movement of the sensor. As a result, it is reasonable to obtain an approximate optimal tour within affordable costs. According to the tests conducted, the recommending value of the initial population size is 100 and the iteration should be in a moderate range from 4×10^4 to 6×10^4.

8.2.5 An exploration of multiple instruments sequencing and path planning

In the case of multiple available instruments and operators, the individual inspection can be carried out simultaneously to enhance the efficiency as well as reducing the cost. The author made an attempt to adopt the concept of multi-travelling salesperson problem to this scenario, which is illustrated in Figure 8.29. With the assistance of modern metrology software, a number of instruments can work collaterally regardless the types and methods of collecting data. Wireless connecting devices such as touch pad and smart mobile are supported by the metrology software. The inspection tasks coupled with the sequence and path guidance are transferred into those devices carried by operators. In the meantime the measurement data from different instruments are acquired by the central computer for subsequent analysis and verification.

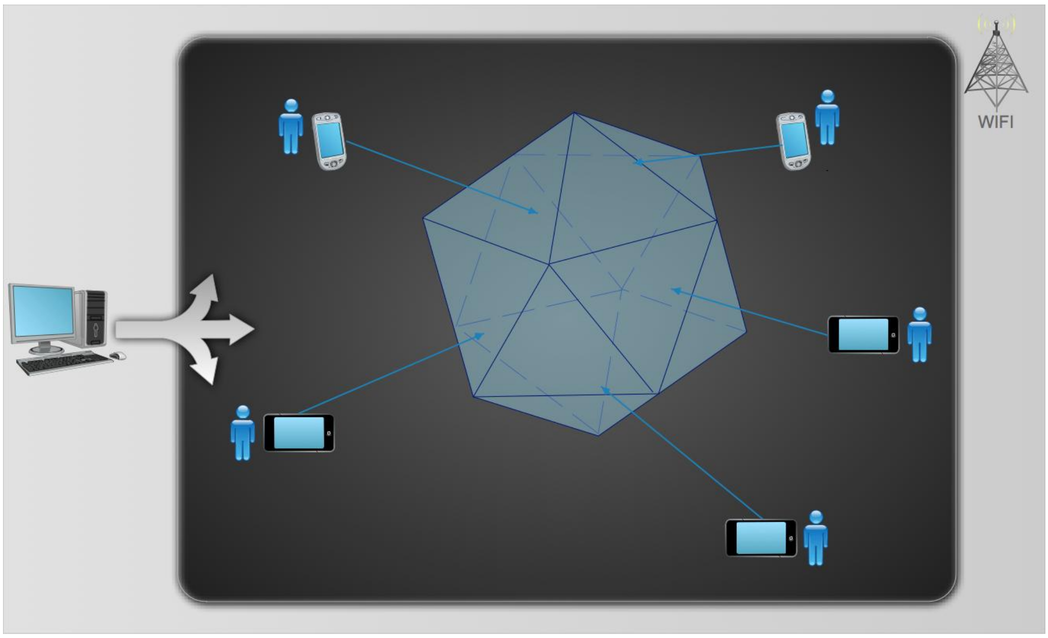


Figure 8.29 A multiple instruments configuration

The proposed genetic algorithm can be accordingly modified to generate the solution for mTSP. By setting up the number of instruments working corporately, and the minimum number of inspection points that each instrument traversed, the optimal path for each individual targeting device is created based on the minimum total travelling distance.

For the example in Figure 8.20, three instruments are available with adequate accessories instead of one and therefore the 30 inspection locations can be covered separately. Each instrument is responsible for 10 locations and the optimal paths for three individual operators are shown in Figure 8.30.

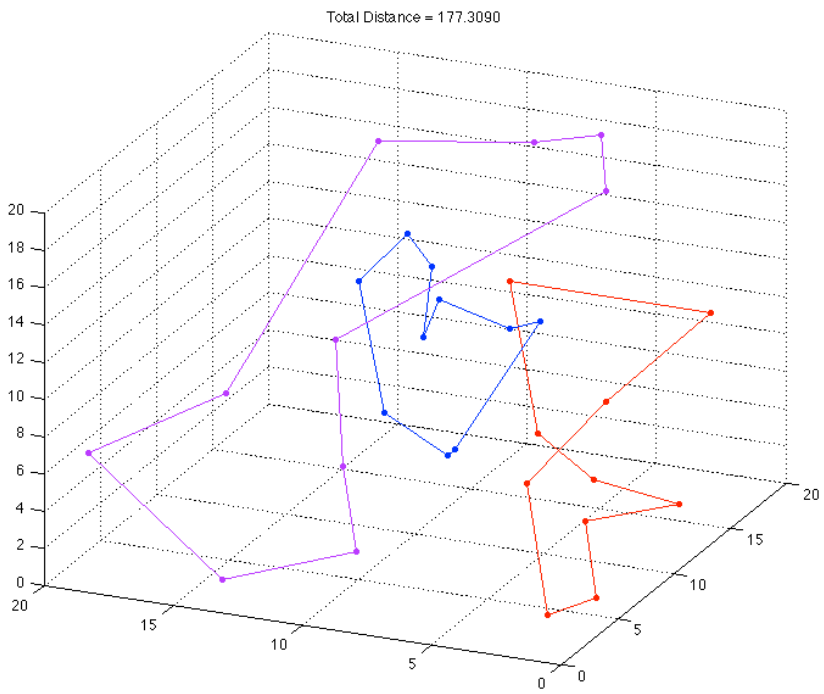


Figure 8.30 Individual optimal paths for three instruments

As far as the sequencing and path planning of inspection points are concerned, multiple instruments can potentially increase the efficiency remarkably. Figure 8.31 shows the optimized paths for three individual operators with the same input of inspection points distribution given in Figure 8.25.

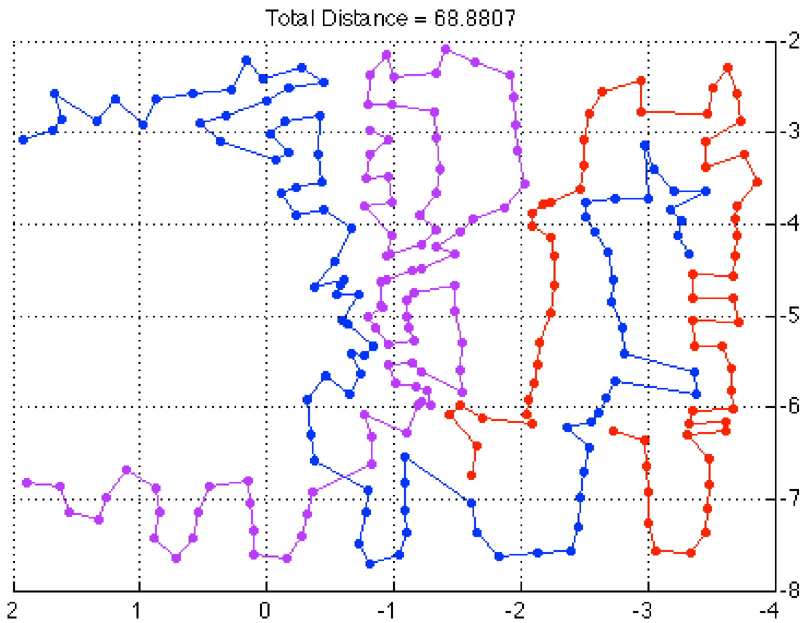


Figure 8.31 Optimized inspection paths for three individual instruments

A minimum number of 70 points is assigned to each operator in this example and the algorithm successfully generates the optimal path with a total length of 68.8807 units, which is a significant decrease compared with the original length of 74.1641 units when only one instrument is employed. In addition, since the three operators conduct the inspection simultaneously, the cost of measuring time is reduced exceptionally.

8.3 Summary

Featuring the ability of mobilizing around the inspection part, the configuration of LVM instruments is flexible and variable. Although the available spatial region is confined by the GICSs, there is still uncertainty regarding the location to place the instrument where the best inspection results are obtained. Under this circumstance, an optimization process is put forward in this chapter aiming to achieve the minimum combined uncertainty. Two approaches are presented for exact solution and discrete solution respectively and examples are given to demonstrate the processes.

Inspection over large-scale product normally features significant traversing distances and this leads to a requirement for sequencing the inspection points with the minimum accumulated length of the paths. In this study, the measurement sequencing process is modelled as a two-level travelling salesman problem (TSP). Instrument positions and locations of inspection points are considered the inputs of two TSPs respectively. A genetic algorithm is developed to efficiently solve the problems approximately regardless of the complexity of the geometry. Examples are given to demonstrate the effectiveness of the approach. An attempt is made to incorporate multiple instruments in order to enhance the efficiency of the inspection process and reduce measurement time.

CHAPTER 9 VALIDATION OF THE DEVELOPED METHODS USING CASE STUDIES

9.1 Introduction

The proposed IPP system, its workflow and functional models have been presented and discussed in previous chapters. Despite the examples given and demonstrated within each module in previous chapters, the effectiveness and functionalities of the system still require comprehensive demonstration and verification by means of case study. This chapter thus exploits the developed system through generating a detailed inspection plan based on the analysis of the design model. The first example consists of the inspection planning process for a basic large-scale part with specified GD&T information. The second example is a typical part from the aerospace industry. The complete workflow of the IPP system is demonstrated via these two examples including inspection task identification, instrument selection and sampling strategy generation, visibility analysis, clustering analysis, instrument configuration and finally measurement sequencing.

9.2 Case study 1: a large-scale part with typical features and GD&T information

In this case study, a large-scale part in the dimension range of 5 meters is presented with typical features such as plane, hole, fillet, and pocket, shown in Figure 9.1. Four tolerances and one dimension are specified using 3D annotation presentation with the CAD model of the part. Three datums are given in this example to construct the reference datum frame. Datum A is the bottom plane that acts as the primary datum and datum B is the perpendicular plane with respect to A on the right of the part, which is secondary datum. Datum C is perpendicular to both A and B, which constructs the tertiary datum. Table 9.1 gives the details regarding the GD&T specification and datums.

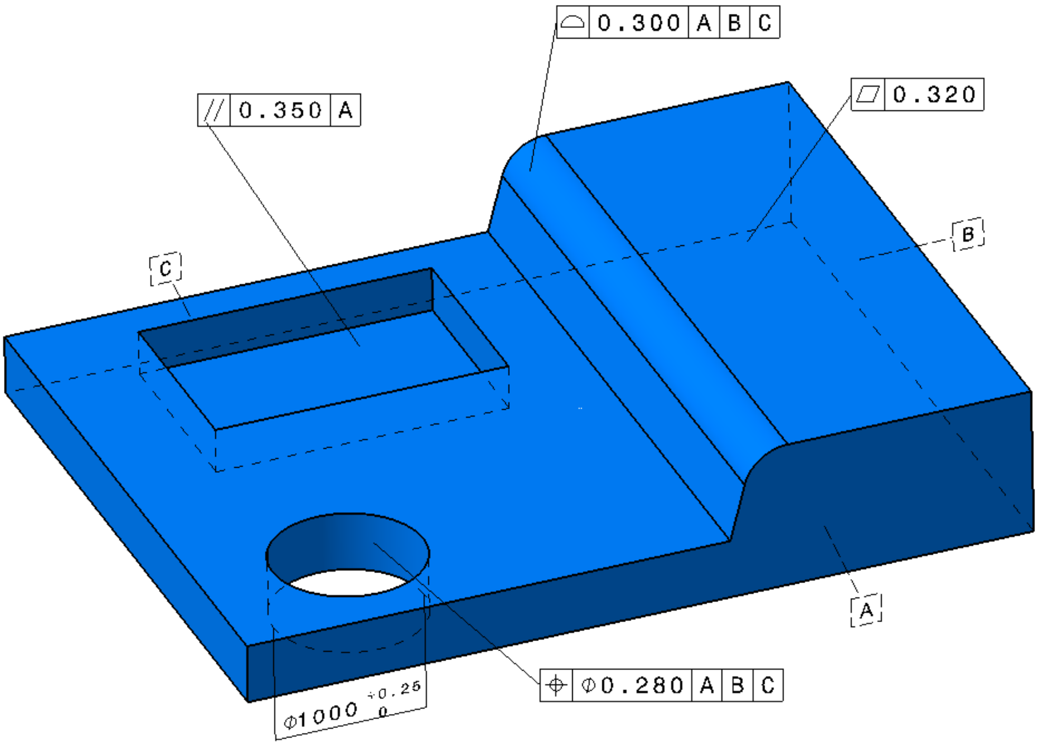


Figure 9.1 An example part with GD&T specification

9.2.1 Tolerance analysis

The CAD model and GD&T specification is firstly imported to SA to conduct the tolerance analysis process. The original description of the GD&T information is shown in Table 9.1 together with corresponding features presented using typical manufacturing feature categories denoted in Figure 9.2. Table 9.2 gives the description of the datums with respect to related features. As described in Chapter 4, each tolerance and dimension are decomposed to present the corresponding feature elements. The results of this process are shown in Table 9.3 and Table 9.4 indicating the decomposed feature elements from tolerances and datums, respectively. Note that the represented features types of the notations are changed after the process.

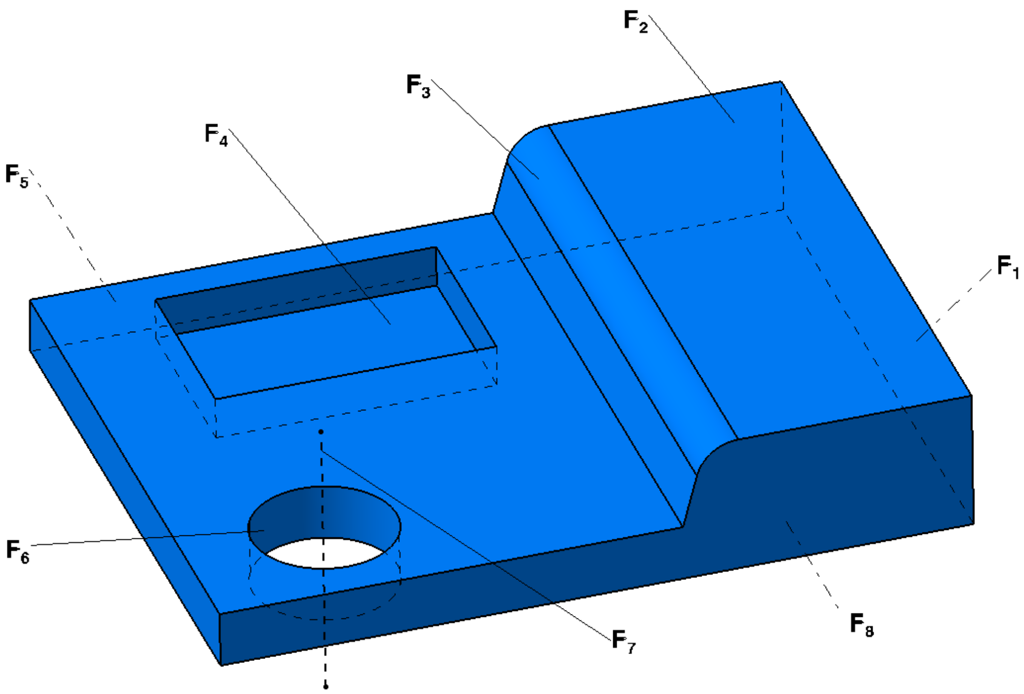


Figure 9.2 Features on the example part

Table 9.1 Description of tolerances and dimension in the example part

Tolerance	Feature	Feature description	Value	Datum
Flatness	F_2	A face on the top	0.320	
Surface Profile	F_3	A fillet face	0.300	A, B, C
Parallelism	F_4	A pocket	0.350	A
Dimension	F_6	A through hole	+0.250	
Position	F_7	A axis	0.280	A, B, C

Table 9.2 Description of datums in the example part

Datum	Feature	Feature element description
A	F_8	A face at the bottom
B	F_1	A face at the right side
C	F_5	A face at the back

Table 9.3 Decomposed GD&T specification with respect to feature elements

Tolerance	Feature element	Feature element description	Datum
Flatness	F_2	A planar surface	
Surface Profile	F_3	A curved surface	A, B, C
Parallelism	F_4	A planar surface	A
Dimension	F_6	A curved surface	
Position	F_7	A line	A, B, C

Table 9.4Decomposed datums with respect to feature elements

Datum	Feature element	Feature element description
A	F_8	A planar surface
B	F_1	A planar surface
C	F_5	A planar surface

By means of surveying the GD&T database within the SA, the IDs of involved feature elements are retrieved and stored. The inspection task list is accordingly generated, presented in Table 9.5.

Table 9.5 Inspection task list

Task_ID	Feature_related object	Feature_element type	Feature ID	Related tolerance	Related datum
1	object_1	surface	F_1		B
2	object_1	surface	F_2	flatness	
3	object_1	curved surface	F_3	surface profile	
4	object_1	surface	F_4	parallelism	
5	object_1	surface	F_5		C
6	object_1	curved surface	F_6	dimension	
7	object_1	line	F_7	position	
8	object_1	surface	F_8		A

9.2.2 Instrument selection and inspection point determination

After the inspection tasks have been identified, the instrument selection module is engaged to choose the most appropriate system to conduct the measurement based on designed preference.

Instrument filtration is initially carried out based on quantitative requirements shown in Table 9.6. In this case study, only one object is involved in the process therefore the stiffness limitation and material property is constant. Moreover, the feature elements are similar in dimension and tolerance value. It is thus feasible and reasonable to assume that all the inspection tasks share the same set of MCs. The uncertainty requirement is determined using a 4 to 1 ratio by the smallest tolerance, which is the dimension of the hole that requires an instrument with the uncertainty capability smaller than 31.25 μm. The range requirement is the largest dimension of the object, which potentially minimizes the number of reconfigurations of the system to complete the tasks. Corresponding crisp MCs are given in Table 9.7.

Table 9.6 Quantitative requirements of the tasks

Task_ID		1-8
Requirements		Details
Environmental conditions	Temperature	22°
	Altitude	300 m
	Humidity	25%
Stiffness limitation		contact& non-contact
Material property		magnet applicable
Uncertainty requirement		31.25 μm
Range		5m

Table 9.7 Crisp MCs of the inspection tasks

Task_ID	1-8
Crisp MCs	Details
$MC_{Temperature}$	22°
$MC_{Altitude}$	300 m
$MC_{Humidity}$	25%
$MC_{Stiffness}$	contact& non-contact
$MC_{Material}$	magnet applicable
$MC_{Uncertainty}$	31.25 μm
MC_{Range}	5m

Available instruments stored in database include a laser tracker, a laser tracker with probe, an articulated arm, a laser radar, an iGPS and a photogrammetry system. The detailed descriptions of corresponding MCs are given in Table 9.8.

Table 9.8 Available inspection systems with associated MCs

Instrument	MC_T (°)	MC_A (m)	MC_H (%)	MC_S	MC_M	MC_U (μm)	MC_R (m)
Laser tracker	0-40	-100-3000	10-90	Non contact	Magnetic	15	80
Laser tracker+ probe	16-24	-100-3000	10-90	Contact	No target	100	18
Articulated Arm	10-28	-100-3000	10-90	Contact	No target	24	3
Photogrammetry	-40-120	-400-3000	10-90	Non contact	Sticker	28	100
Laser radar	5-40	-400-3000	10-90	Non contact	No target	24	60
iGPS(4 series)	10-30	-400-3000	10-75	Non contact	Magnetic	200	40*40

The filtration process is implemented as follows:

- Articulated arm is removed due to insufficient range coverage.
- laser tracker with probe and iGPS are filtered out due to unsatisfied uncertainty requirement.

Under this circumstance, laser tracker, photogrammetry system and laser radar have remained from Phase 1, as the alternative instruments

I_1: Laser tracker

I_2: Photogrammetry system

I_3: Laser radar

One designer (DM_1) and two metrologists (DM_2 and DM_3) are involved in the performance evaluation process based on the four fuzzy MCs：

C_1: Instrument uncertainty

C_2: Overall cost

C_3: Inspection Speed

C_4: TRL

The process of fuzzy instrument selection then takes place. The three DMs are assigned a different importance factor and the calculated weights are given in Table 9.9.

Table 9.9 DMs importance and corresponding weights

	DM_1	DM_2	DM_3
Linguistic importance	Important	Medium	Very important
Weight	0.356	0.238	0.406

Each instrument is subsequently evaluated by each DM with respect to the four fuzzy criteria and the aggregated decision matrix is shown in Table 9.10. The importance for all fuzzy criteria is shown in Table 9.11.

Table 9.10 Performance rating of instruments

Criteria	Instrument	DM_1	DM_2	DM_3
C_1 uncertainty	I_1	EG	EG	VG
	I_2	F	G	G
	I_3	G	G	VG
C_2 cost	I_1	VH	VH	H
	I_2	M	H	M
	I_3	EH	EH	EH
C_3 speed	I_1	VG	F	F
	I_2	EG	VG	VG
	I_3	VG	EG	VG
C_4 TRL	I_1	VG	EG	VG
	I_2	VG	VG	VG
	I_3	F	G	B

Table 9.11 Assigned importance for all fuzzy criteria

Criteria	DM_1	DM_2	DM_3
C_1	VI (0.90,0.10)	I (0.75,0.20)	VI (0.90,0.10)
C_2	VI (0.90,0.10)	VI (0.90,0.10)	M (0.50,0.45)
C_3	I (0.75,0.2)	M (0.50,0.45)	VI (0.90,0.10)
C_4	M (0.50,0.45)	I (0.75,0.20)	M (0.50,0.45)

Having gathered the above information, the instrument selection process is able to generate the rank list using the process described in Chapter 5 and the relative closeness coefficients and rank are given in Table 9.12. Therefore the laser tracker is selected to conduct this inspection assignment.

Table 9.12 Relative closeness coefficient and rank of individual instrument

Instruments	RC	Rank
I_1	0.639	1
I_2	0.525	2
I_3	0.503	3

As the laser tracker is a point-based measurement system, the sampling strategy on the part is therefore to use discrete points on each feature element. Followed by the standards and recommendation given in Chapter 5, the distribution of inspection points are assigned and indicated in Figure 9.3. Only points on viewable surfaces are given in the figure for better illustrative purpose. The number of points of each feature element is given in Table 9.13. Note that F_7 is constructed based on the measured cylinder and therefore no actual inspection is carried out.

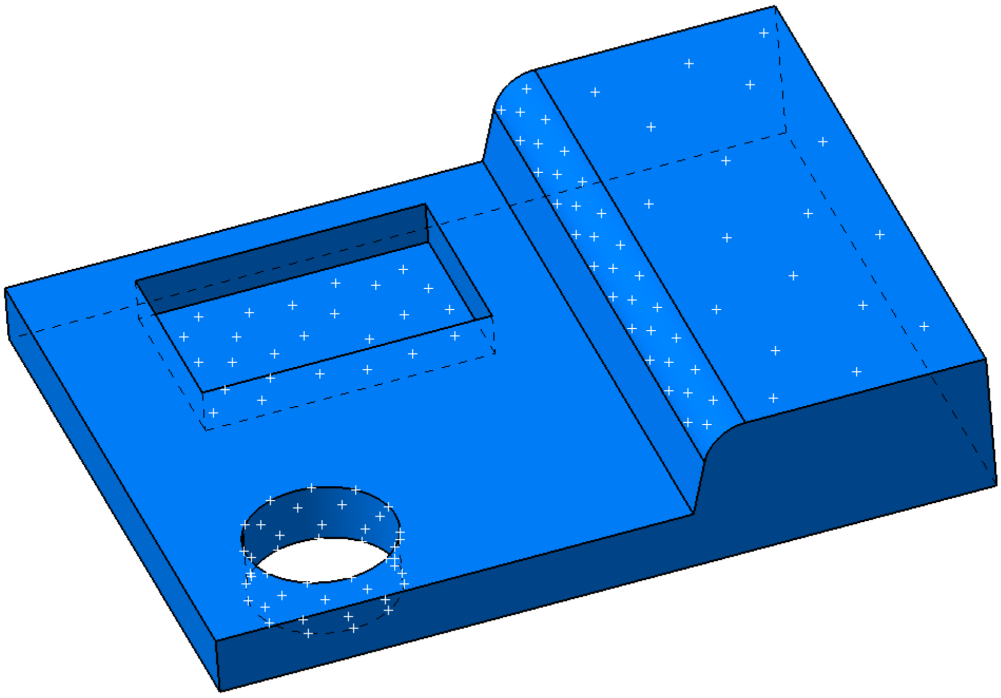


Figure 9.3 Points distribution on feature elements

Table 9.13 Number of points on each feature element

Task_ID	Feature ID	Feature_element type	Number of Points
1	F_1	surface	28
2	F_2	surface	18
3	F_3	curved surface	30
4	F_4	surface	25
5	F_5	surface	40
6	F_6	curved surface	44
7	F_7	line	
8	F_8	surface	30

9.2.3 Visibility analysis

The first step of visibility analysis is to identify the convex and concave area and the results are given in Figure 9.4 where red regions denote the concave area while the blue regions denote the convex area. F_1, F_2, F_5, and F_8 are in the convex area and F_3, F_4, and F_6 are in the concave area. Consequently, the GICSs on the convex area are equal to the LICSs, which can be obtained by generating the MSs at the measuring points. In this example the measurable space of the laser tracker is set at 10 meters since it is adequate to cover the entire object and it is not appropriate to locate the instrument further than this MS as the uncertainty is proportional to the distance between the target and the instrument. Similarly, only the centre inspection point on each feature is analysed to represent the visibility region of the entire feature such that computational cost can be reduced.

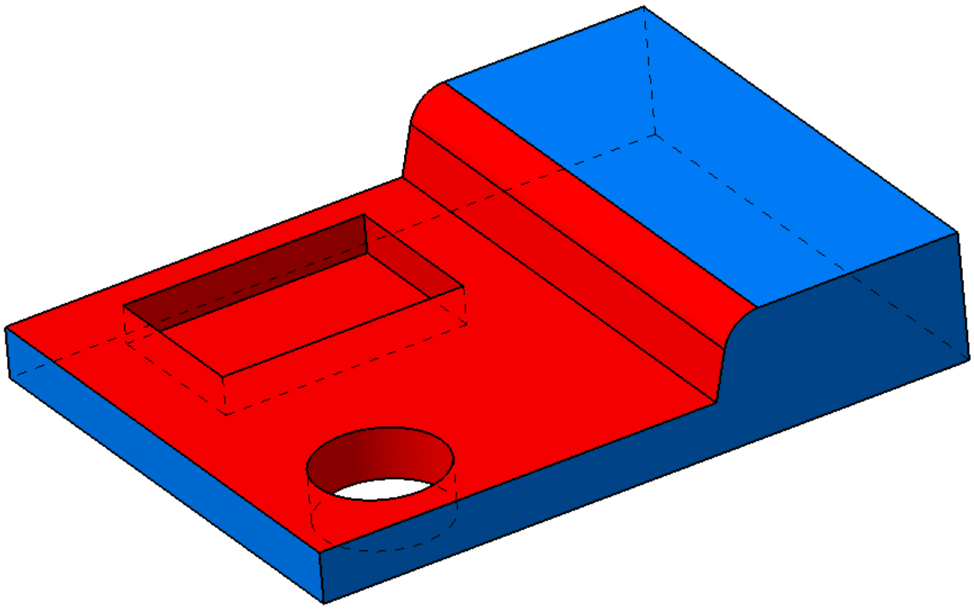
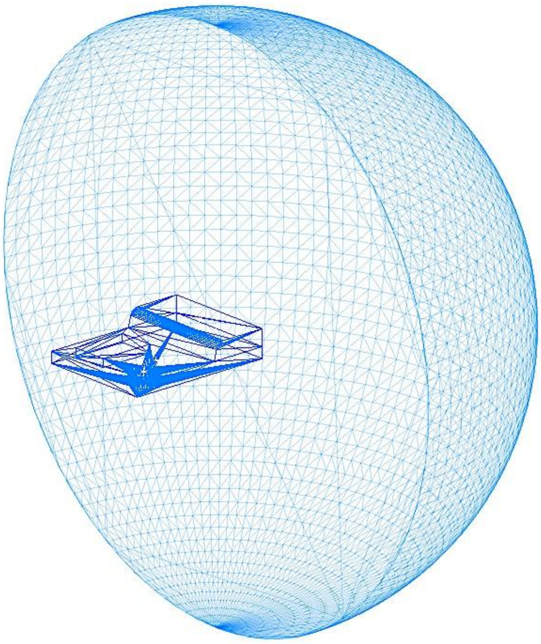
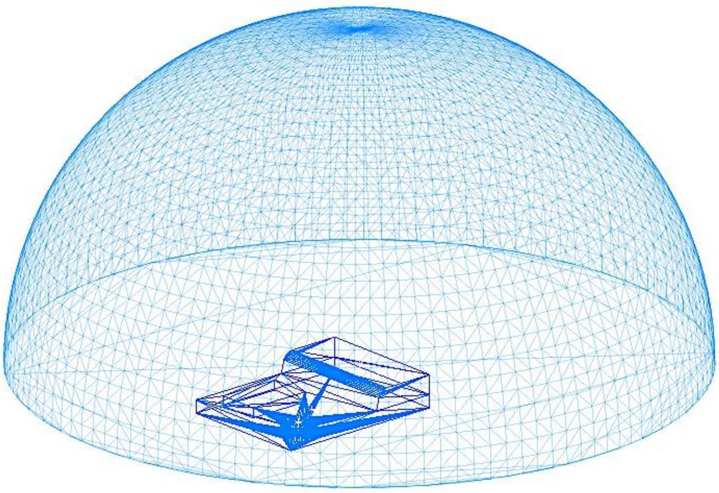


Figure 9.4 Concave and convex region on the part

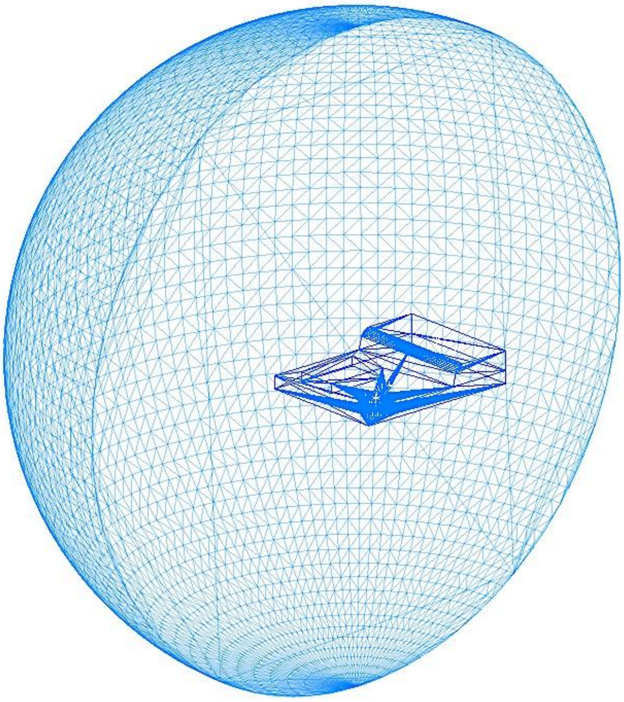
The GICSs of features in the convex region are then created accordingly and shown in Figure 9.5 (a)-(d).



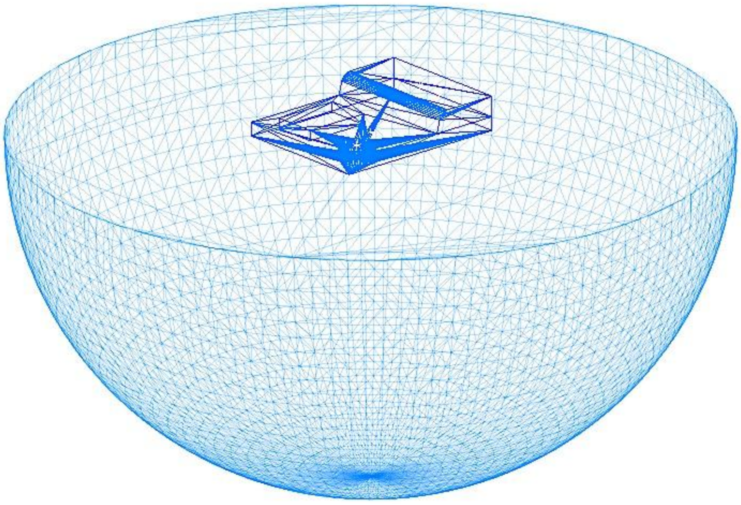
(a) *GICS(F_1)*



(b) *GICS(F_2)*



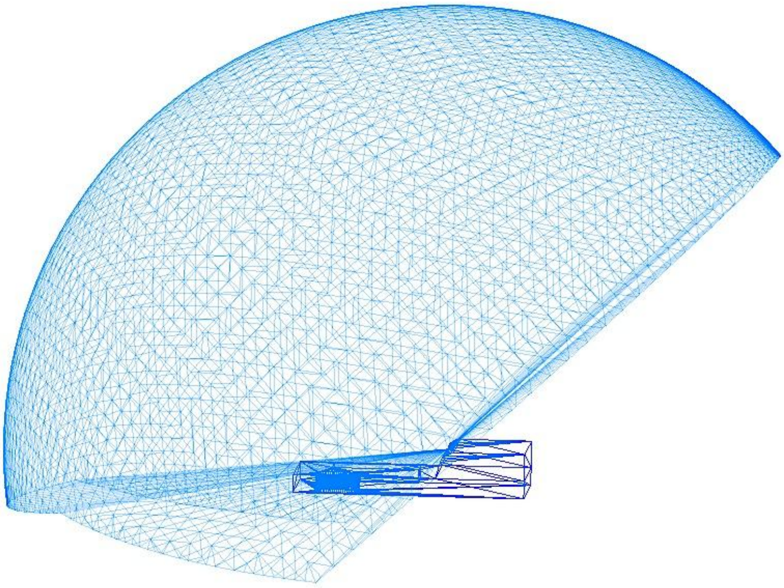
(c) *GICS(F_5)*



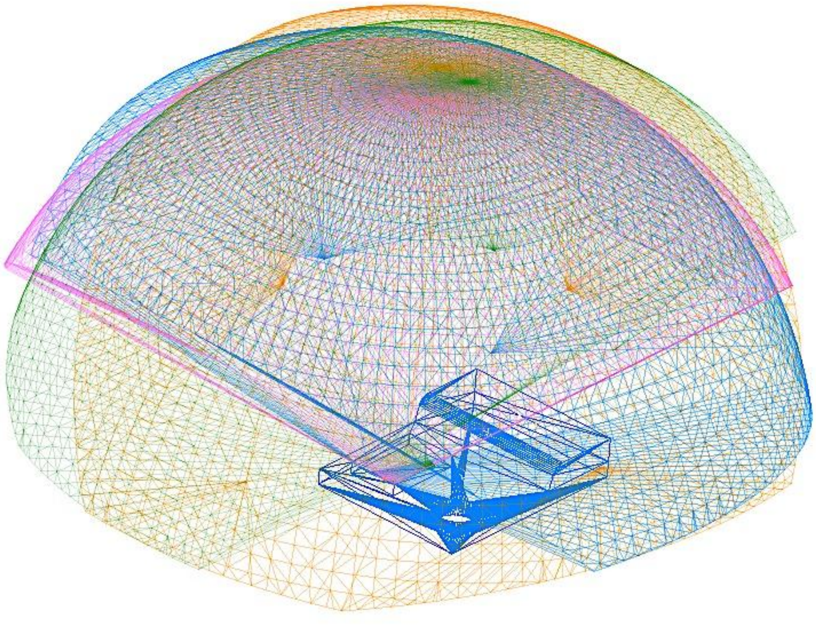
(d) *GICS(F_8)*

Figure 9.5 Global instrument configuration spaces for convex features

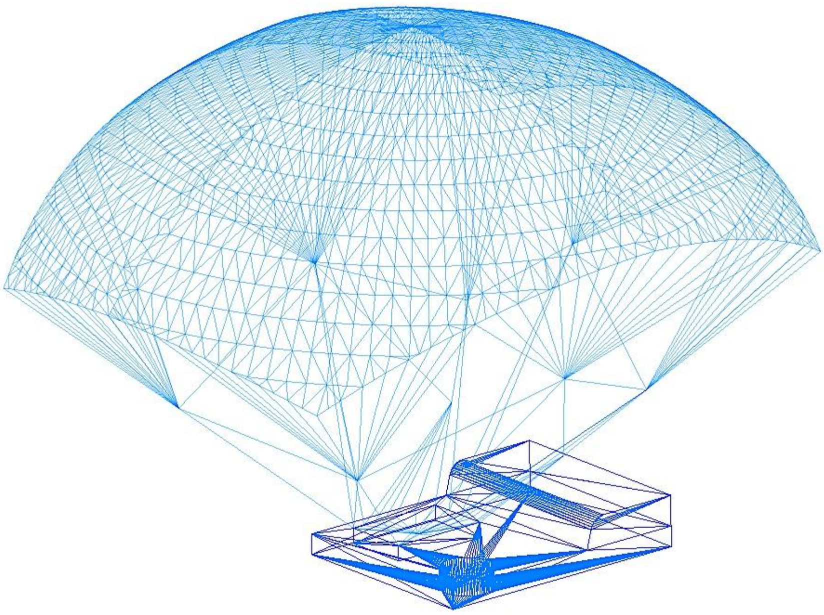
For features on the concave surfaces, the GICSs are obtained in Figure 9.6 (a)-(d) using the process presented in Chapter 6. Inspection points on each feature are grouped in this example and the GICSs within every group are intersected before the clustering analysis to save computational cost. To illustrate this process, Figure 9.6(b) shows the GICSs of four boundary points on F_4 and Figure 9.6(c) shows the aggregated configuration space.



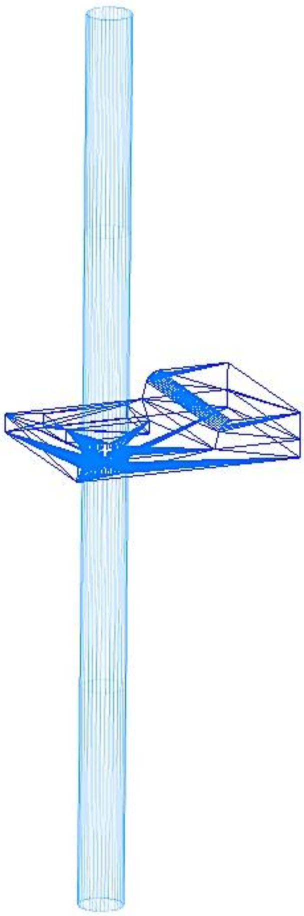
(a) $GICS(F_3)$



(b) GICSs on feature F_4



(c) *GICS(F_4)*



(d) *GICS(F_6)*

Figure 9.6 Global instrument configuration spaces for concave features

9.2.4 Clustering analysis

The outputs of the visibility analysis are the GICSs of all features. As inspection points are bounded by their associated features, the clustering analysis is carried out based on the feature elements in this case. The distribution of all seven GICSs is shown in Figure 9.7 and the clustering algorithm described in Chapter 7 is conducted to determine the intersection that contains the most inspection features.

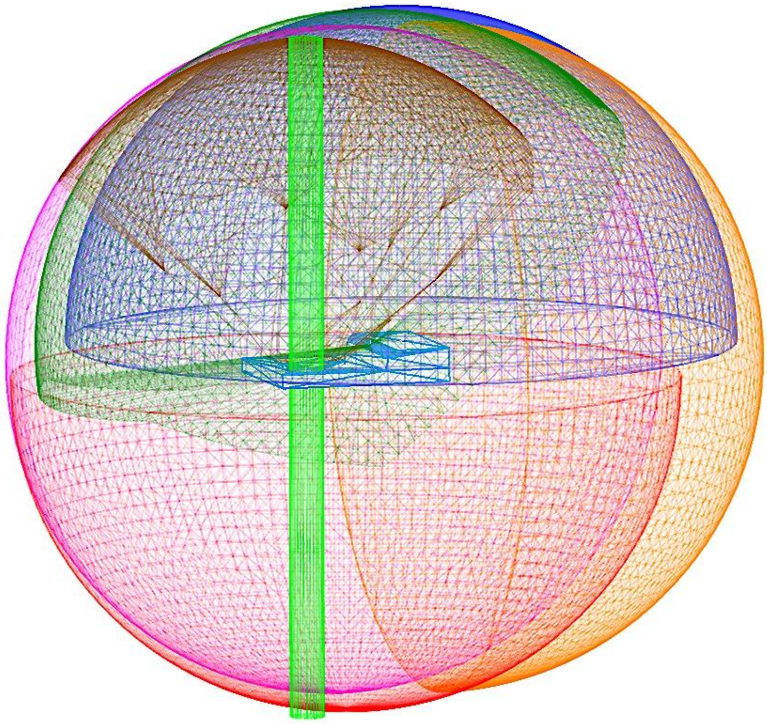


Figure 9.7 Distribution of all GICSs

The universe A is consisted of seven features $\{F_1, F_2, F_3, F_4, F_5, F_6, F_8,\}$ and five unique subsets are found based on the intersection results:

$S_1 = \{F_1, F_2, F_3, F_4, F_5\}$

$S_2 = \{F_1, F_2, F_3, F_6\}$

$S_3 = \{F_1, F_5, F_8\}$

$S_4 = \{F_2, F_3, F_4, F_6\}$

$S_5 = \{F_3, F_5, F_8\}$

$S_6 = \{F_6, F_8\}$

The corresponding binary matrix is then constructed as:

$$\begin{array}{c} \\ F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \\ F_8 \end{array}
\begin{array}{cccccc} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 \\ \hline \end{array}$$

	S_1	S_2	S_3	S_4	S_5	S_6
F_1	1	1	1	0	0	0
F_2	1	1	0	1	0	0
F_3	1	1	0	1	1	0
F_4	1	0	0	1	0	0
F_5	1	0	1	0	1	0
F_6	0	1	0	1	0	1
F_8	0	0	1	0	1	1

The set cover is then obtained $C = \{S_1, S_6\}$ using the greedy heuristic and Figure 9.8 gives the intersection results based on the clustering strategy.

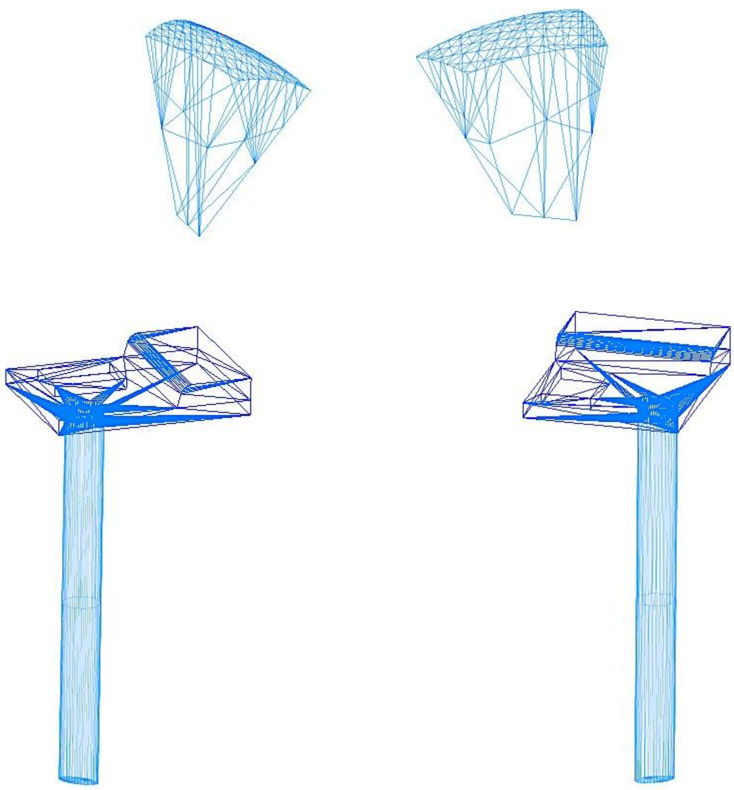
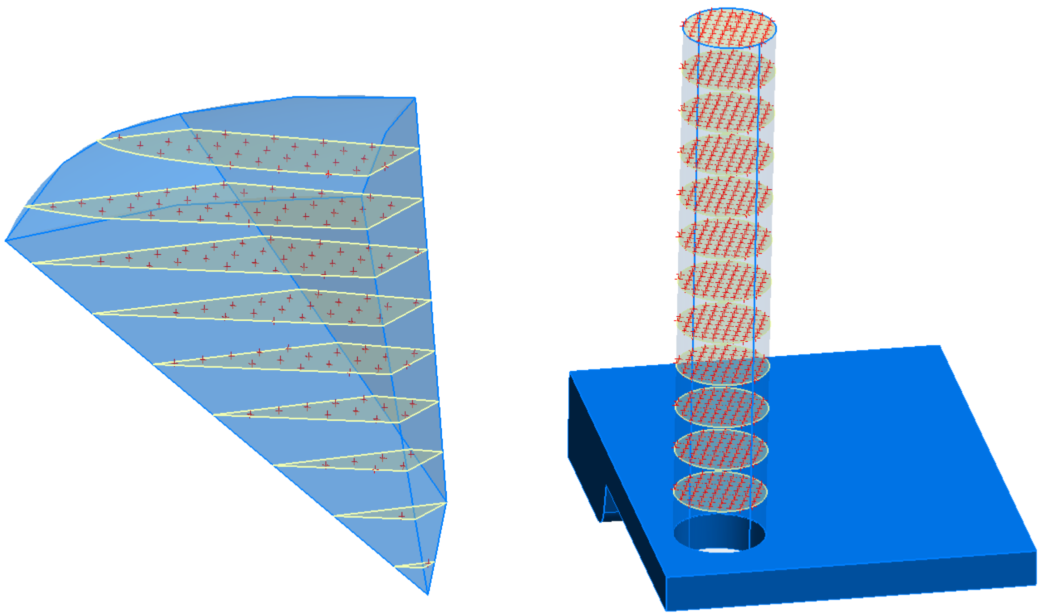


Figure 9.8 Clustering analysis results showing the final clusters in two viewing directions

9.2.5 Instrument setup and configuration

Discrete solution is adopted in this case study in order to determine the optimal positions of the instrument within the GICSs, which have the minimum combined uncertainty to all inspection points.

The final *GICS*(S_1) is discretized at the step of 500mm and the sampling points are 500mm away with each other, as shown in Figure 9.9(a). As the primary source of uncertainty for laser tacker is from the distance, one can estimate that the performance is maximized when the location of the system is the nearest to all the inspection points. Therefore *GICS*(S_6) shown in Figure 9.9(b) is only sampled for half of the volume instead of the entirely region and the distance between each pair of points in the same sampled surface is 100mm since the variation of location within the surface is more important.



(a) GICS(S_1) (b) GICS(S_6)

Figure 9.9 Discretized GICSs and sampling points

The measurement simulation is conducted subsequently by locating the instrument at each sampled point with respect to all viewable inspection points. Having compared all the combined uncertainties by querying for the point that yields the minimum algebraic sum of individual uncertainties, the best location of the instrument is generated. Table 9.14 shows the details of the simulated measurements at one alternative location with respect to the 18 points on Feature 2.

Table 9.14 Simulated measurements of 18 inspection points on F_2

	Inspection points on Feature 2						
Point Name	**X**	**Y**	**Z**	**Ux**	**Uy**	**Uz**	**Mag**
p0	-2776.5957	1818.9452	6011.0454	0.0327	0.0325	0.0269	0.0534
p1	-2386.0545	1724.0095	6434.2258	0.034	0.0337	0.0269	0.0549
p2	-1831.7197	1812.5786	6820.4558	0.0344	0.0347	0.0265	0.0556
p3	-2265.0493	1382.4438	6865.0841	0.0351	0.0357	0.0284	0.0576
p4	-2773.7531	1454.0974	6363.7358	0.0343	0.0345	0.0271	0.0557
p5	-3108.8391	1041.7291	6474.8729	0.0354	0.0344	0.0275	0.0565
p6	-2626.1743	1119.7224	6810.4068	0.0348	0.0351	0.0277	0.0567
p7	-2239.5728	962.8778	7289.6918	0.0378	0.0376	0.027	0.0598
p8	-2665.918	613.017	7263.2606	0.0375	0.0386	0.028	0.0606
p9	-3053.1227	613.0295	6934.0182	0.036	0.0367	0.0305	0.0598
p10	-3474.9535	290.2869	6885.135	0.0343	0.0376	0.0284	0.0583
p11	-3032.2303	166.0966	7380.9621	0.0375	0.0402	0.0294	0.0624
p12	-2616.2728	147.7669	7752.3306	0.0396	0.0409	0.0298	0.0643
p13	-2972.8768	-279.0861	7859.0075	0.0407	0.0415	0.0312	0.066
p14	-3443.9973	-249.5767	7429.9983	0.0382	0.0397	0.0314	0.0634
p15	-3020.1384	-831.461	8349.1748	0.0413	0.0421	0.0319	0.0671
p16	-3459.0922	-836.6691	7981.0299	0.0407	0.0422	0.0337	0.0676
p17	-3891.551	-727.0749	7507.9928	0.039	0.04	0.0311	0.064

The optimal locations in both GICSs are eventually obtained and the results are illustrated in Figure 9.10 together with the instrument model.

9.2.6 Measurement sequencing

As there are only two instrument locations existing in this particular example, the optimization of instrument locations is unnecessary. Measurement sequencing is conducted for the two clusters S_1 and S_6 respectively. 145 inspection points in S_1 are modelled as the TSP using 3D Euclidean distance (Figure 9.11 (a)) and the proposed GA algorithm is carried out to generate the approximately optimal solution shown in Figure 9.11(b), which has the total length of 43970mm. The population size is set to 120 and the goal is achieved at 57162th iteration. The solution history is given in Figure 9.11(c).

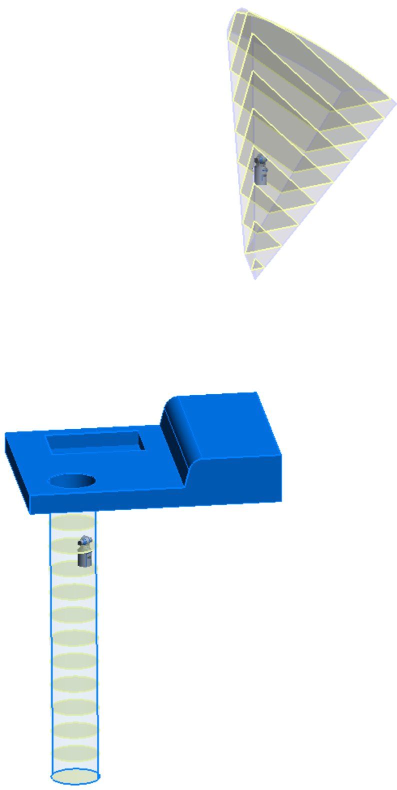
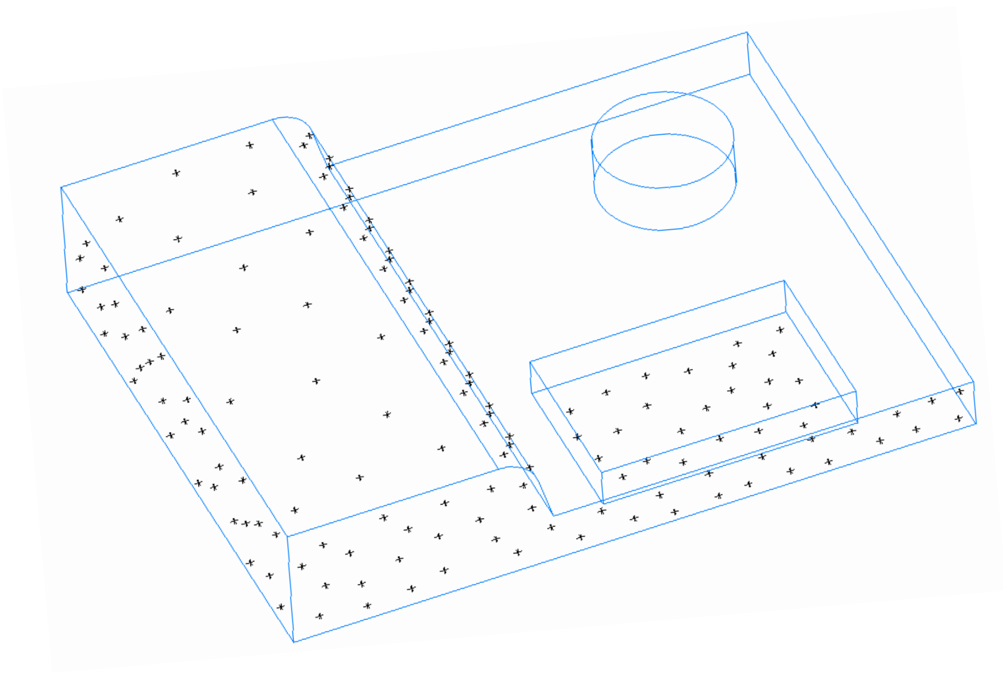
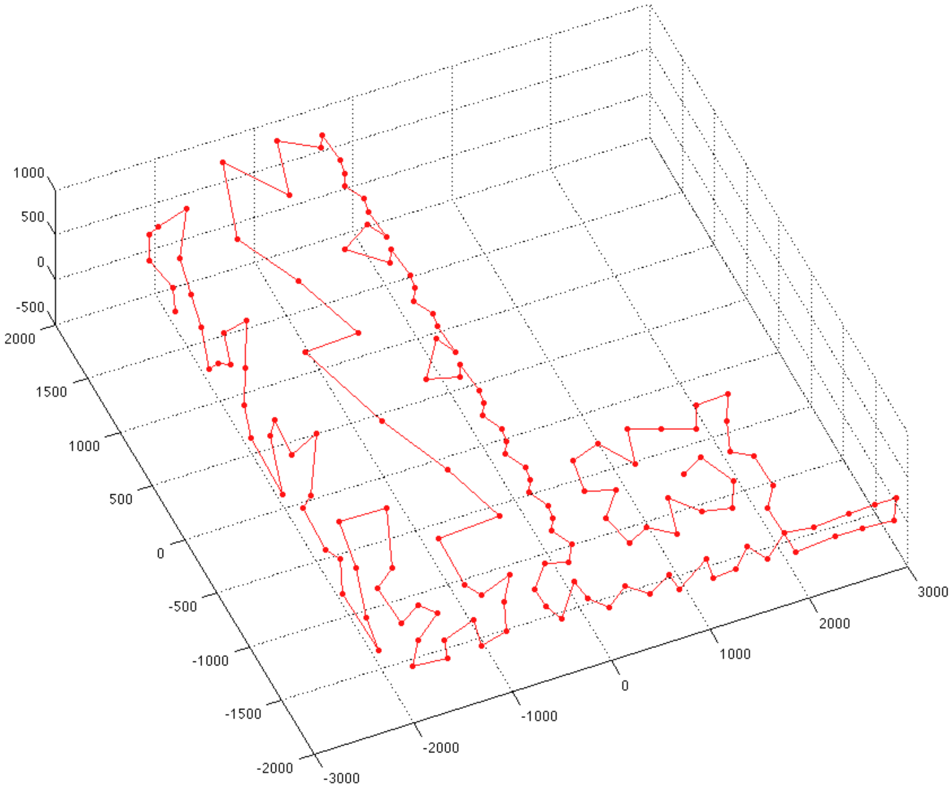


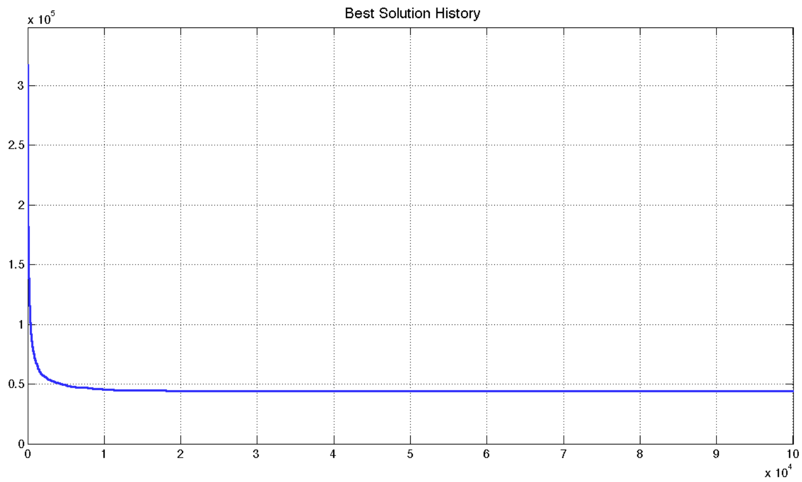
Figure 9.10 Optimal locations of the instrument in *GICS*(S_1) and *GICS*(S_6)



(a) Distribution of inspection points in S_1



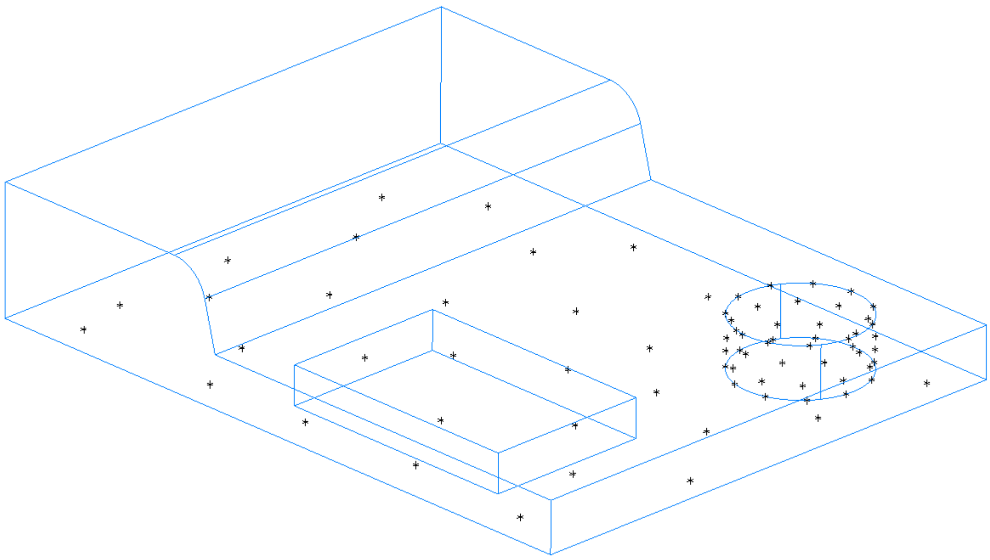
(b) Approximately optimal measuring sequence



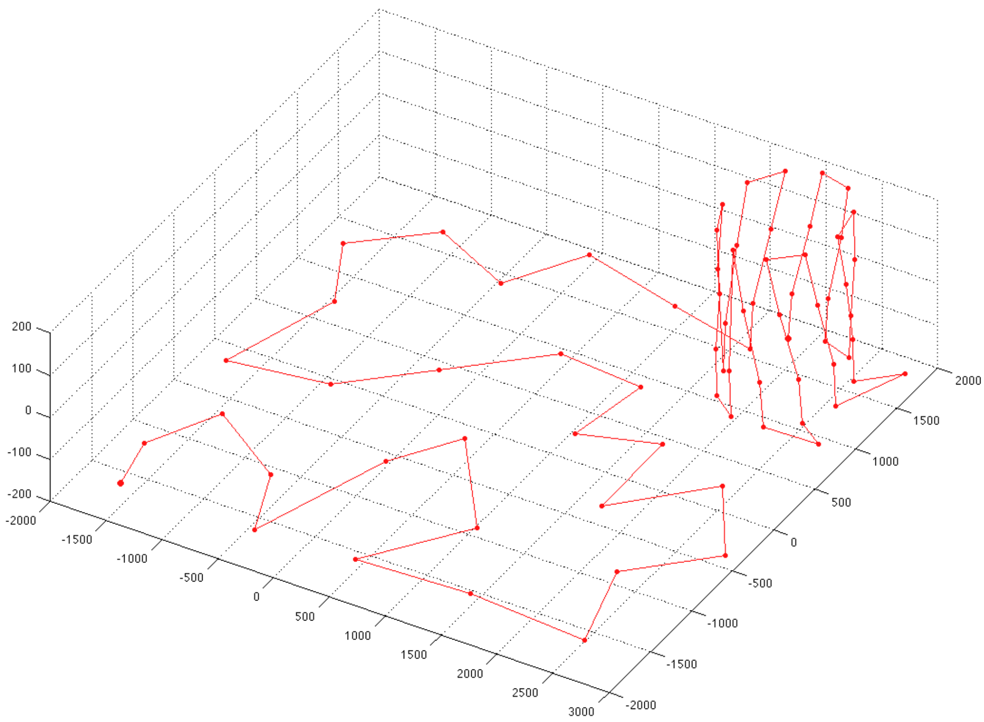
(c) Solution history

Figure 9.11 TSP modelling and solution of S_1

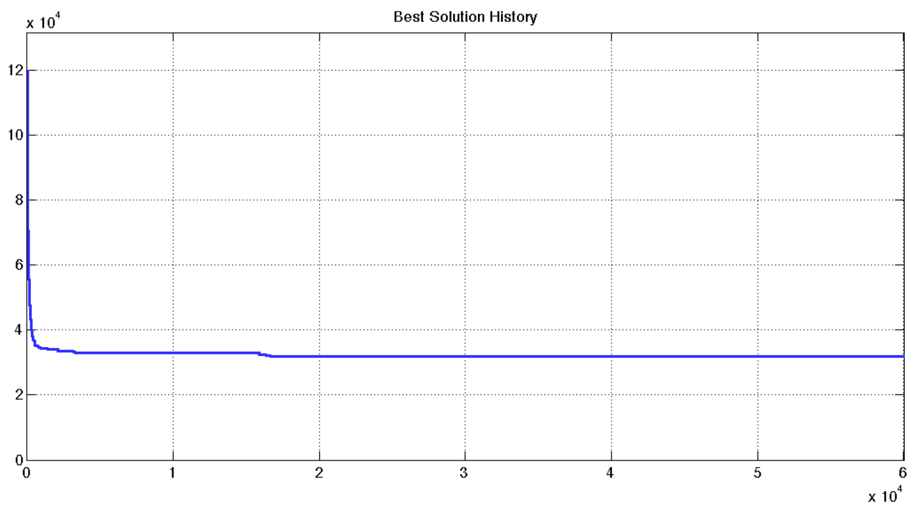
74 inspection points in S_2 are modelled as the TSP using 3D Euclidean distance (Figure 9.12 (a)) and the generated approximately optimal solution is shown in Figure 9.12(b), which has the total length of 43970mm. The population size is set to 120 and the computation is completed at 57162th iteration. Figure 9.12(c) indicates the solution history.



(a) Distribution of inspection points in S_6



(b) Approximately optimal measuring sequence



(c) Solution history

Figure 9.12 TSP modelling and solution of S_6

The generated inspection sequence is then utilized to guide the metrologists to conduct the measurements of all the points using the SMR and 3D coordinates of those points can be accordingly obtained, which enable the geometry fitting and GD&T check process within SA.

9.3 Case study 2: an assembly wing box with given inspection points

The proposed framework is tested in this section on an aerospace industrial part to validate the methods. An assembly wing box is presented with pre-defined inspection points as well as the distribution, shown in Figure 9.13. The dimension of the part is about 1500mm ×2000mm×1500mm. This is a common scenario in product validation process where inspection points are determined according to best practice or empirical plan generated from historical products. Therefore this case study is valid to demonstrate the adaptation and the flexibility of the framework to different prerequisites and inputs.

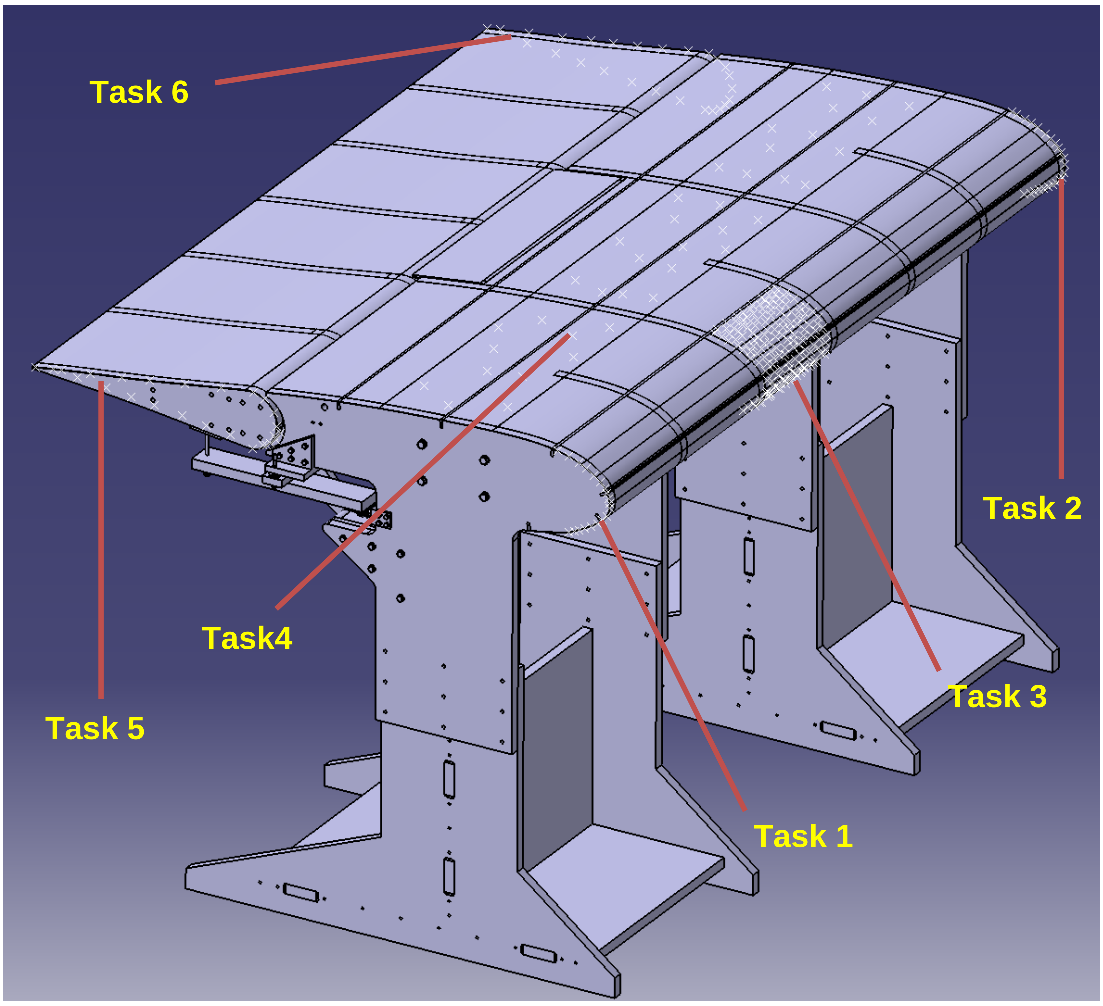


Figure 9.13 The wing box with inspection points

There are six inspection tasks on this part including:

(1) T_1: the profile of the leading edge at the right end

(2) T_2: the profile of the leading edge at the left end

(3) T_3: one free-form surface on the leading edge

(4) T_4: the flat surface on the top

(5) T_5: the profile of the flap at the right end

(6) T_6: the profile of the flap at the left end

The inspection tasks are carried out by acquiring the 3D coordinates of all the points and then compared with the CAD model to check the deviations. GD&T information is not needed. The framework is thus implemented from the instrument selection stage.

9.3.1 Instrument selection

The quantitative requirements of the inspection are given with the inspection tasks in Table 9.15 and corresponding crisp MCs are given in Table 9.16.

Table 9.15 Quantitative requirements of the tasks

Task_ID		1-8
Requirements		Details
Environmental conditions	Temperature	22°
	Altitude	1000 m
	Humidity	35%
Stiffness limitation		contact& non-contact
Material property		magnet applicable
Uncertainty requirement		90 μm
Range		3m

Table 9.16 Crisp MCs of the inspection tasks

Task_ID	1-8
Crisp MCs	Details
$MC_{Temperature}$	22°
$MC_{Altitude}$	1000 m
$MC_{Humidity}$	35%
$MC_{Stiffness}$	contact& non-contact
$MC_{Material}$	magnet applicable
$MC_{Uncertainty}$	90 μm
MC_{Range}	3m

The instrument database is given in Table 9.8 in Section 9.2.2 except that the photogrammetry and the laser tracker with probe is not available to this inspection job. The filtration process is then conducted as follows:

- The iGPS is filtered out due to unsatisfied uncertainty requirement.

Therefore, the laser tracker, the photogrammetry system and the laser radar have remained from Phase 1, as the alternative instruments

I_1: Laser tracker

I_2: Articulated arm

I_3: Laser radar

One designer (DM_1) and two metrologists (DM_2 and DM_3) are involved in the performance evaluation process based on the four fuzzy MCs：

C_1: Instrument uncertainty

C_2: Overall cost

C_3: Inspection Speed

C_4: TRL

The three DMs are assigned with different importance factors and the calculated weights are given in in Table 9.17.

Table 9.17 DMs importance and corresponding weights

	DM_1	DM_2	DM_3
Linguistic importance	Very important	Important	Medium
Weight	0.406	0.356	0.238

Fuzzy evaluation of each instrument is taken place by each DM against four fuzzy criteria and the aggregated decision matrix is shown in Table 9.18. The importance for all fuzzy criteria is shown in Table 9.19. Given the nature of this inspection, the validation process should be confident in accuracy while the cost is reasonable and the instrument must be reliable and robust to meet the production requirements.

Table 9.18 Performance rating of instruments

Criteria	Instrument	DM_1	DM_2	DM_3
C_1 uncertainty	I_1	EG	G	VG
	I_2	G	G	F
	I_3	G	G	G
C_2 cost	I_1	H	VH	H
	I_2	M	M	M
	I_3	EH	EH	EH
C_3 speed	I_1	G	G	F
	I_2	F	VG	VG
	I_3	VG	VG	G
C_4 TRL	I_1	VG	EG	EG
	I_2	VG	VG	VG
	I_3	F	F	F

Table 9.19 Assigned importance for all fuzzy criteria

Criteria	DM_1	DM_2	DM_3
C_1	VI (0.90,0.10)	VI (0.90,0.10)	VI (0.90,0.10)
C_2	VI (0.90,0.10)	VI (0.90,0.10)	I (0.75,0.2)
C_3	I (0.75,0.2)	M (0.50,0.45)	I (0.75,0.2)
C_4	VI (0.90,0.10)	VI (0.90,0.10)	I (0.75,0.2)

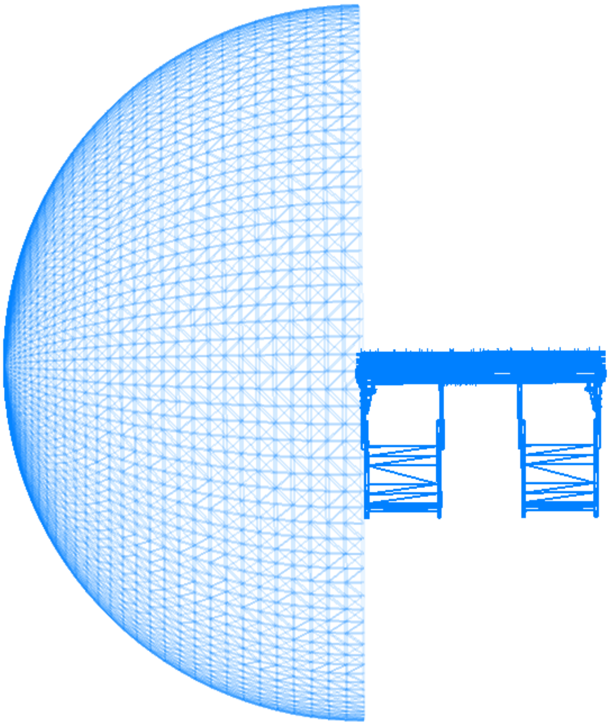
With the above input to the fuzzy selection process, the ranking method shown in Chapter 5 is carried out and the relative closeness coefficients are given in Table 9.20. The laser tracker is selected to be the most suitable instrument to conduct this inspection assignment. The score of the articulated arm is very close to the laser tracker but the laser radar performs insufficiently due to the low TRL

Table 9.20 Relative closeness coefficient and rank of individual instrument

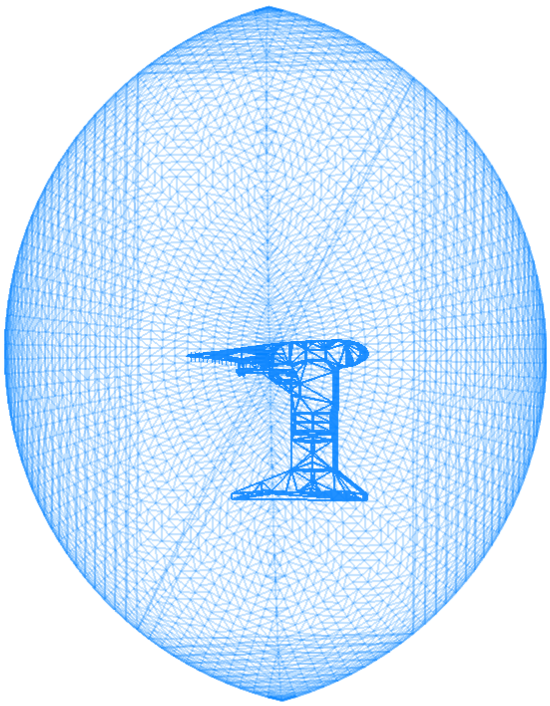
Instruments	RC	Rank
I_1	0.601	1
I_2	0.584	2
I_3	0.403	3

9.3.2 Visibility analysis

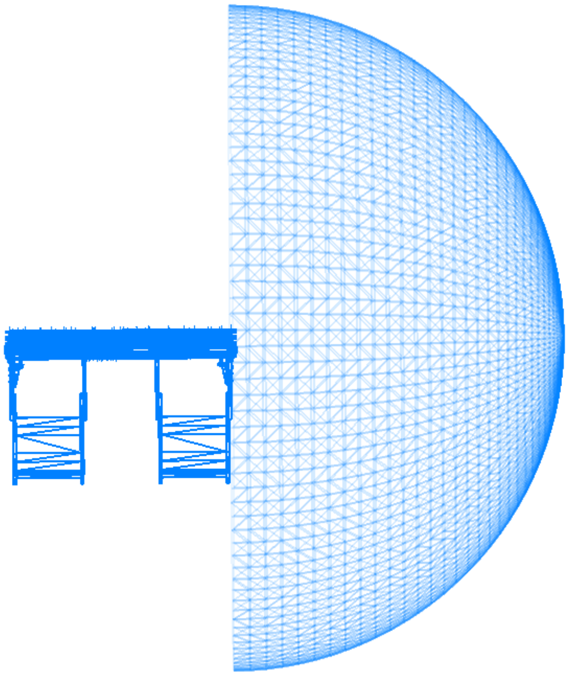
It is assumed that the inspection of T_1, T_2, T_5 and T_6 must be carried out from the side of the wing to obtain traceable results. Therefore T_1, T_2, T_4, T_5 and T_6 are all on the convex area of the part and their GICSs can be generated by constructing LICSs at the measurement points. In order to minimise the uncertainty of laser tracker, the MS is set at 3 meters to cover the entire part. *GICS(T_1)* and *GICS(T_5)* are aggregated before the clustering process as T_1 and T_5 are on the same convex surface. The clustering process will be simplified significantly by reducing the subsets. The same process is carried out for *GICS(T_2)* and *GICS(T_6)*. The GICSs of surfaces on the convex region are shown in Figure 9.14 (a)-(f). On the contrast, T_3 consists of six surfaces and three of them are in the concave region. The LICSs of T_3 are given in Figure 9.15(a) using different colours and the aggregated *GICS(T_3)* is shown in Figure 9.15(b).



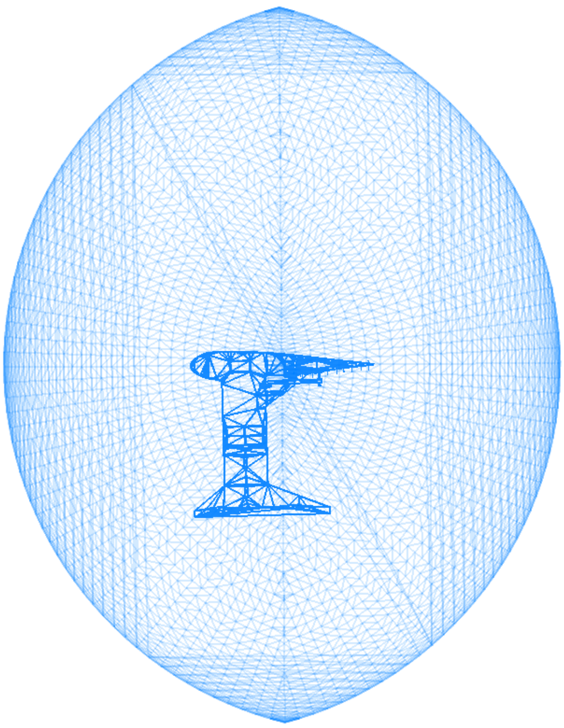
(a) *GICS(T_1, T_5)* from front view



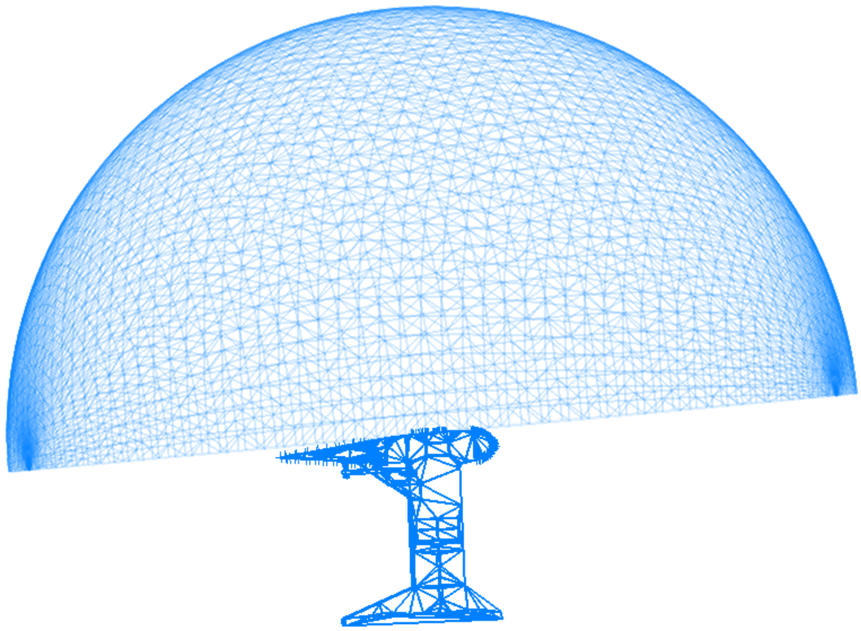
(b) *GICS(T_1, T_5)* from side view



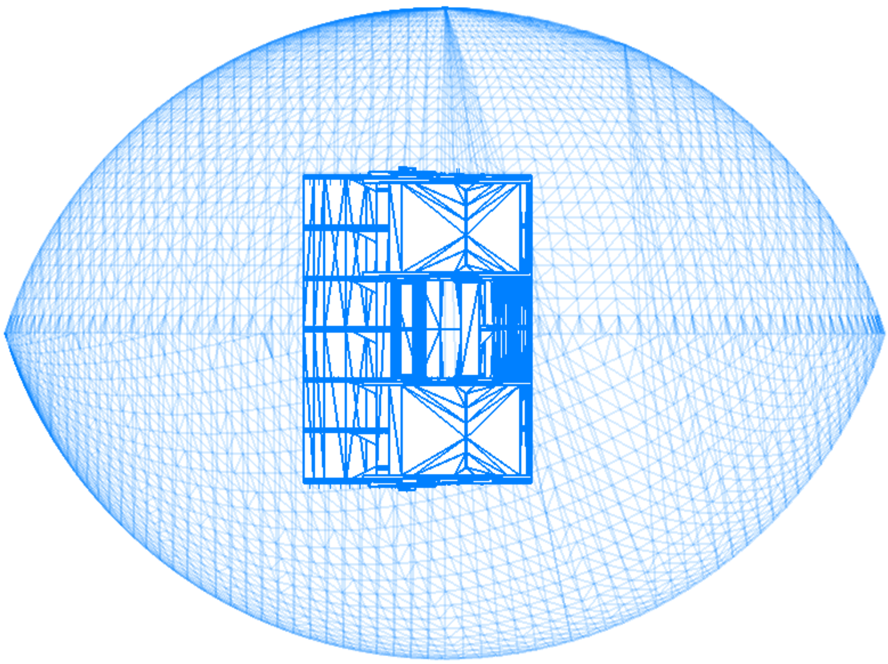
(c) *GICS(T_2, T_6)* from front view



(d) *GICS(T_2, T_6)* from side view

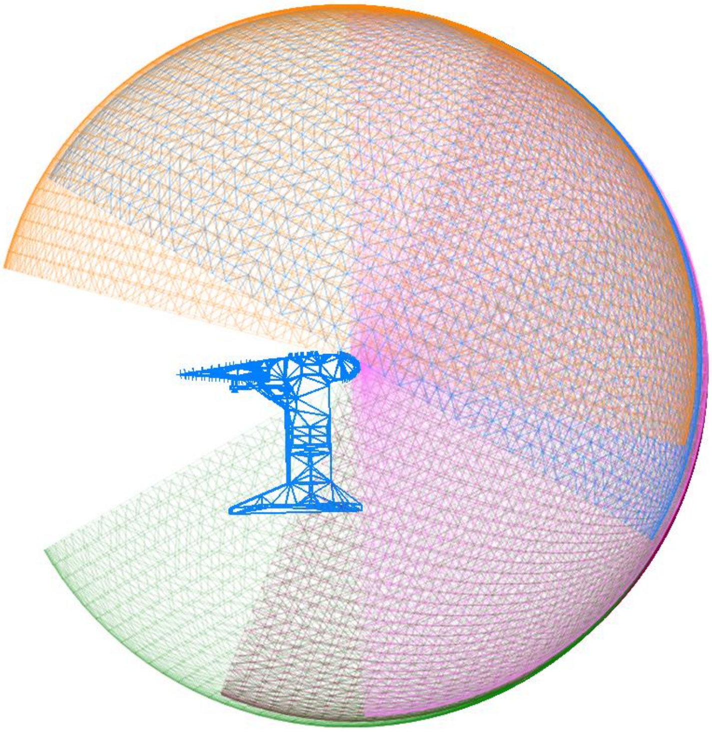


(e) *GICS(T_4)* from side view

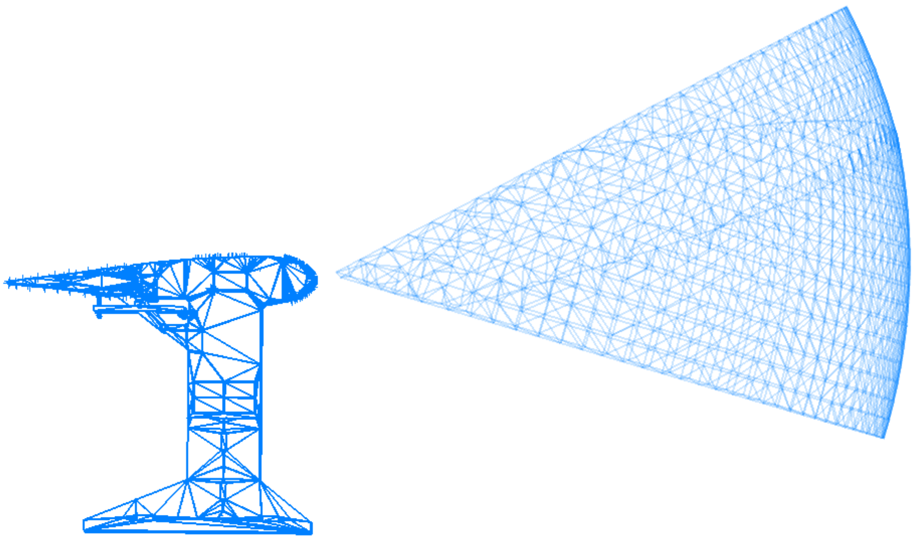


(f) *GICS(T_4)* from top view

Figure 9.14 Global instrument configuration spaces for convex features



(a) *LICSs(T_3)*



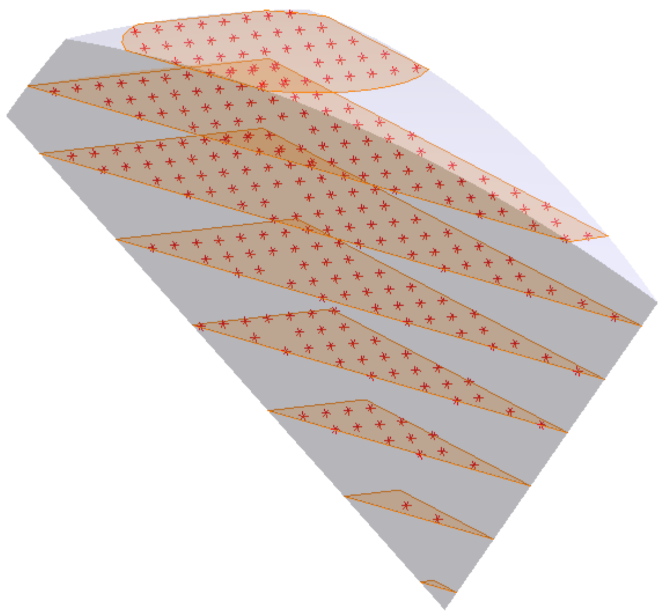
(b) *GICS(T_3)*

Figure 9.15 Global instrument configuration spaces for concave surfaces

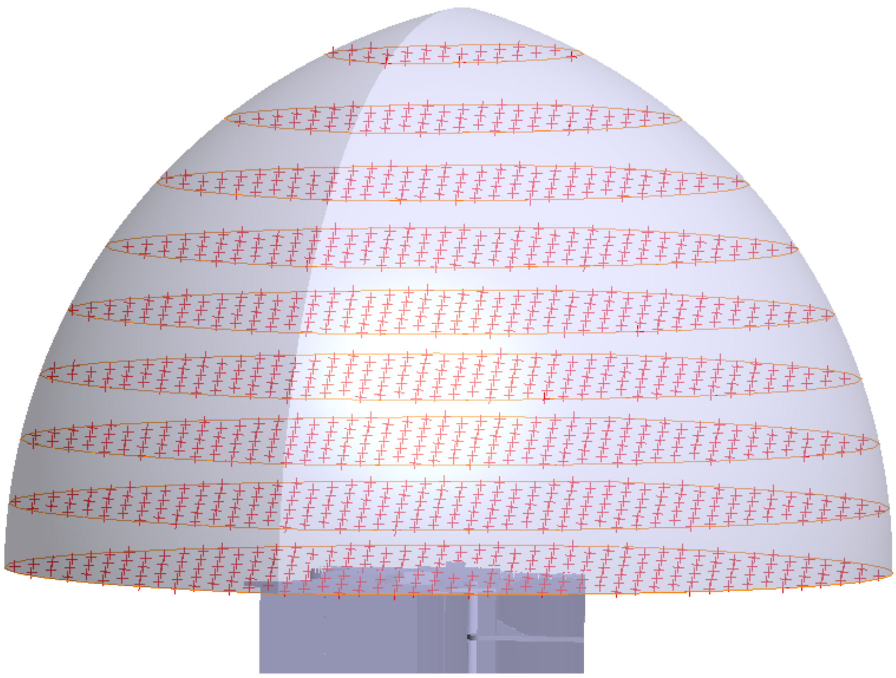
9.3.3 Instrument setup and configuration

Discrete solution is selected in this case study in order to determine the optimal positions of the instrument within the aggregated GICSs, which have the minimum combined uncertainty to all inspection points.

The final $GICS(S_1)$ is discretized at the step of 130mm and the sampling points are 100mm away with each other, as shown in Figure 9.16(a). $GICS(S_9)$ is discretized at the step of 300mm and the sampling points are 300mm away with each other, as shown in Figure 9.16(b). The measurement simulation is conducted subsequently by locating the instrument at each sampled point with respect to all viewable inspection points. The optimal locations in both GICSs are obtained subsequently and the results are illustrated in Figure 9.17 (a)-(c) together with the laser tracker and tripod. The optimal locations are obtained by comparing all the combined uncertainties that are the minimum algebraic sum of individual uncertainties. As an example, Table 9.21 shows the fabricated measurements of 35 inspection points of T_4 at the optimal location with simulated uncertainties.



(a) Discretized $GICS(S_1)$



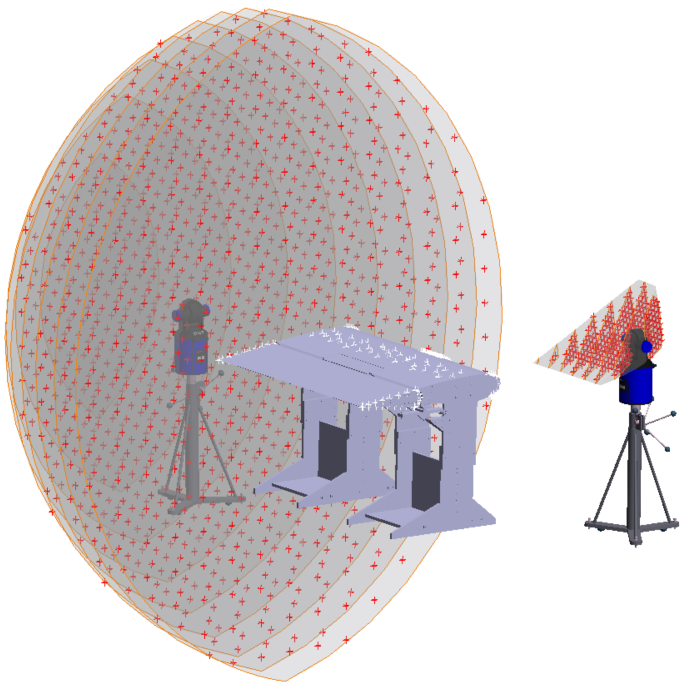
(b) Discretized *GICS(S*$_9$*)*

Figure 9.16 Discretized GICSs and sampling points

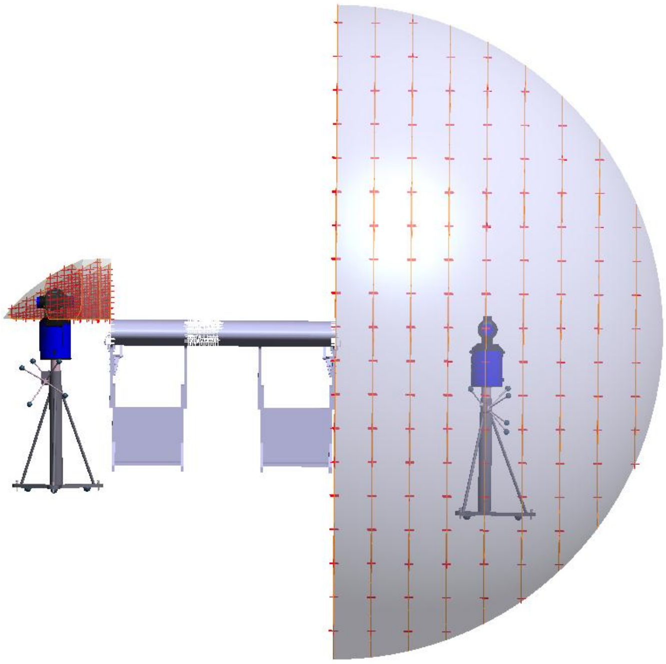
Table 9.21 Simulated measurements of 35 inspection points of T_4 at the optimal location

Simulated uncertainty for inspection points of T_4							
Point Name	**X**	**Y**	**Z**	**Ux**	**Uy**	**Uz**	**Mag**
p0	-509.614	1275.5984	-575.8338	0.013	0.0125	0.0135	0.0225
p1	-490.3692	1276.7576	-820.3776	0.0138	0.0135	0.0146	0.0243
p2	-377.4608	1282.4674	-796.0895	0.0134	0.0135	0.0142	0.0237
p3	-393.8257	1281.2407	-442.9309	0.0126	0.0117	0.0131	0.0216
p4	-507.9155	1275.7313	-151.2408	0.0123	0.0113	0.0119	0.0205
p5	-379.7337	1282.3849	-146.2492	0.0119	0.011	0.0124	0.0204
p6	-560.0578	1272.2057	-0.665	0.012	0.0104	0.0119	0.0198
p7	-379.2764	1281.8937	78.8587	0.0115	0.0098	0.0114	0.0189
p8	-473.2853	1277.6972	61.0952	0.0118	0.0102	0.0117	0.0195
p9	-484.7276	1277.051	-430.443	0.0128	0.0121	0.0129	0.0219
p10	-542.2589	1273.476	225.3823	0.0121	0.0103	0.011	0.0193
p11	-337.9668	1283.5047	232.88	0.0108	0.0092	0.0106	0.0177
p12	-360.0183	1282.7006	-623.833	0.013	0.0126	0.0133	0.0225
p13	-558.036	1272.3421	-667.0081	0.0136	0.0135	0.0139	0.0236
p14	-519.6736	1274.9927	619.4397	0.0114	0.009	0.0096	0.0174
p15	-501.3352	1276.0983	440.2322	0.0113	0.0091	0.0103	0.0178

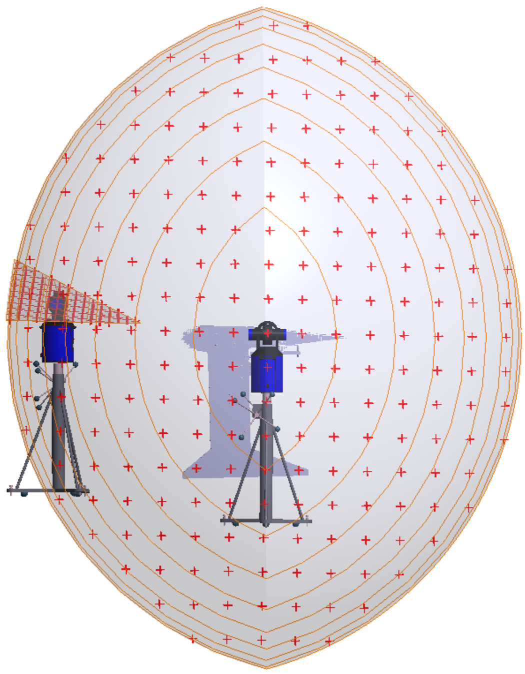
p16	-347.2561	1283.6743	409.4383	0.0108	0.0087	0.0101	0.0171
p17	-397.5072	1281.5904	486.4198	0.011	0.0086	0.0097	0.017
P18	-336.7327	1284.0577	584.5461	0.0105	0.0082	0.009	0.0161
P19	-387.2769	1282.0296	721.2713	0.0112	0.0083	0.0091	0.0167
P20	-335.2283	1284.1121	876.6761	0.0109	0.0076	0.0083	0.0156
P21	-559.9006	1272.224	856.9258	0.0118	0.0085	0.0089	0.0171
P22	-460.4215	1277.8985	922.1854	0.0114	0.008	0.0083	0.0162
P23	-475.2507	1277.0964	727.5137	0.011	0.0087	0.0091	0.0167
P24	-570.8586	1271.3885	513.8811	0.0116	0.0094	0.0102	0.0181
P25	-454.2014	1278.7171	256.9192	0.0111	0.0094	0.0107	0.018
P26	-335.6903	1284.0933	-226.4167	0.0119	0.0109	0.0124	0.0203
P27	-341.6405	1283.8871	-899.4062	0.0136	0.0137	0.0141	0.0239
P28	-337.9542	1283.4839	-435.3422	0.0122	0.0119	0.013	0.0214
P29	-560.6186	1272.185	-911.9497	0.0143	0.0143	0.0147	0.025
P30	-456.8825	1278.5949	-674.0768	0.0132	0.0128	0.0138	0.023
P31	-560.382	1271.6907	-436.7798	0.0127	0.0123	0.013	0.0219
P32	-559.0284	1271.7884	-257.6854	0.0127	0.0117	0.0129	0.0215
P33	-458.0864	1278.5201	-911.3644	0.0139	0.0138	0.0147	0.0245
P34	-453.9005	1278.7372	-261.0698	0.0122	0.0112	0.0128	0.021



(a) Oblique view of the optimal locations of the laser tracker in *GICS(S_1)* and *GICS(S_9)*



(b) Front view of the optimal locations of the laser tracker in *GICS(S_1)* and *GICS(S_9)*

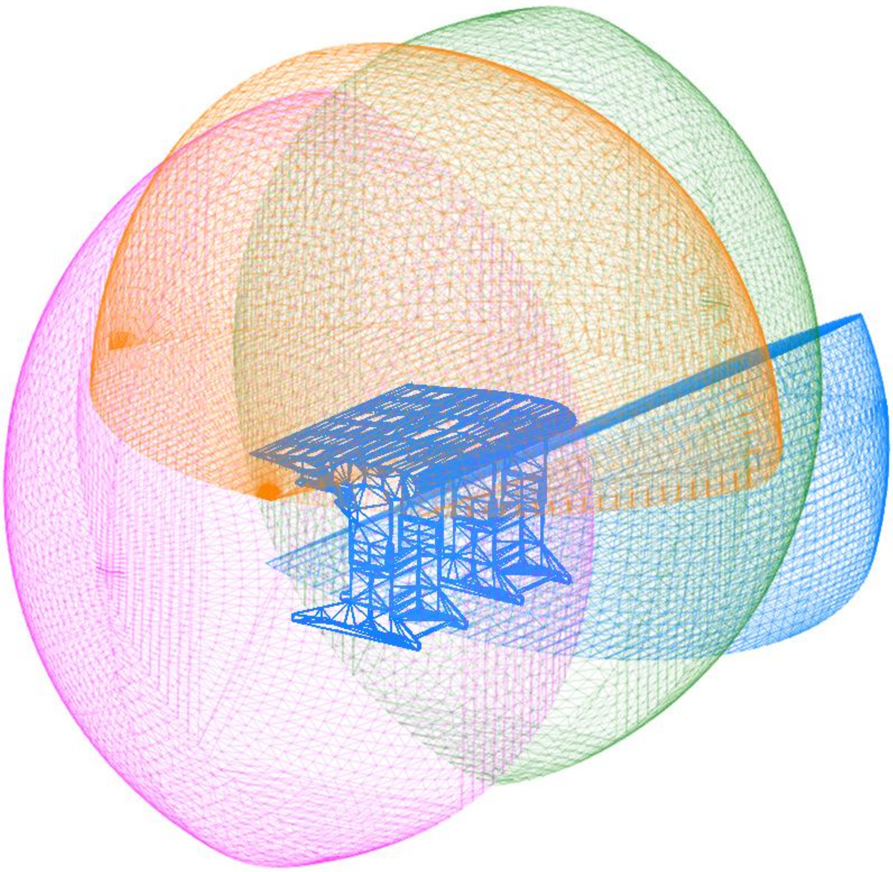


(c) Side view of the optimal locations of the laser tracker in *GICS(S_1)* and *GICS(S_9)*

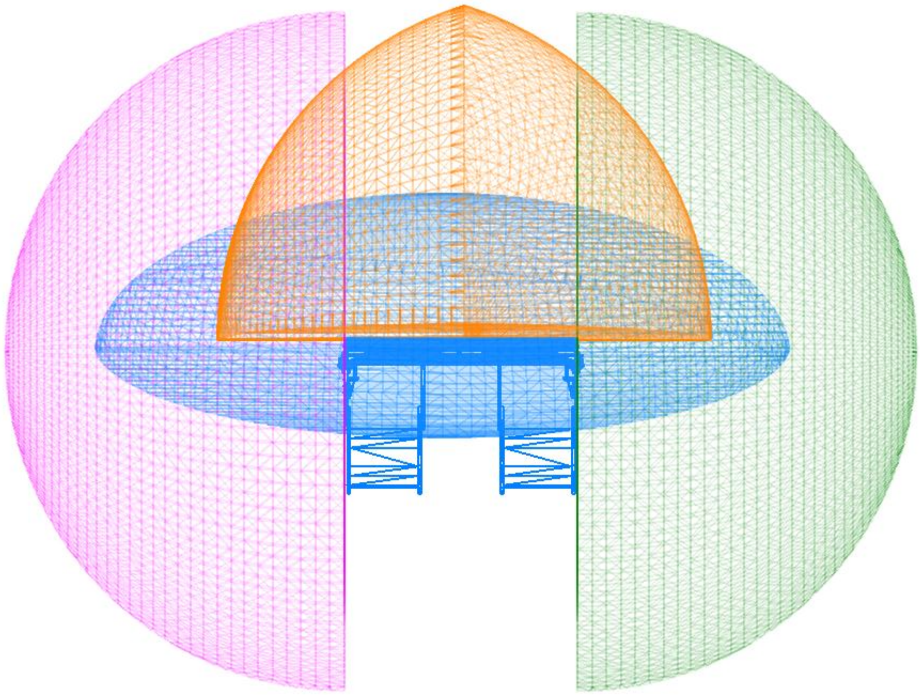
Figure 9.17 Optimal locations of the laser tracker in *GICS(S_1)* and *GICS(S_9)*

9.3.4 Clustering analysis

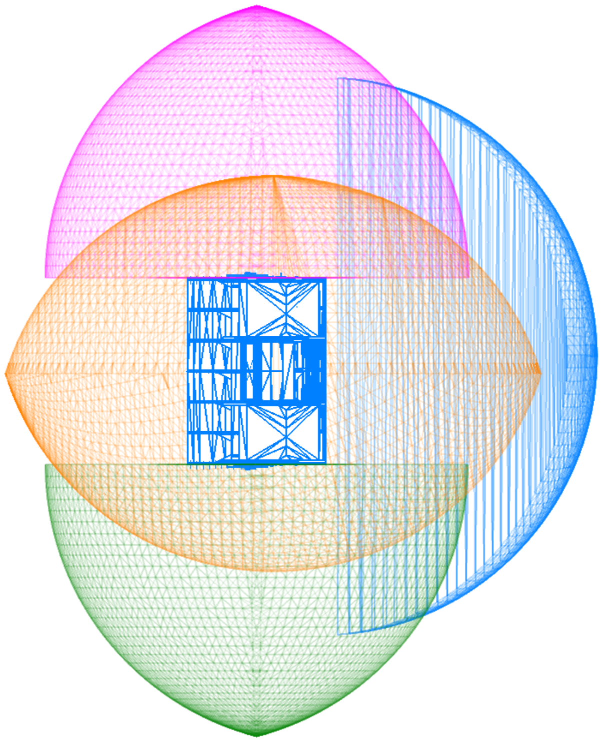
GICSs of all surfaces are generated during the visibility analysis and then inputted into clustering analysis for finding the minimum intersections that cover all inspection points. The distribution of all seven GICSs is shown in Figure 9.18 (a)-(c) and the clustering algorithm described in Chapter 7 is conducted to determine the intersection that contains the most inspection features.



(a) Oblique view of all GICSs



(b) Front view of all GICSs



(c) Top view of all GICSs

Figure 9.18 Distribution of all GICSs in different views

As the GICSs for T1, T5 and T2, T6 have been aggregated in the visibility analysis, the universe A is consisted of four tasks $\{T_{1,5}, T_{2,6}, T_3, T_4\}$ and five unique subsets are found based on the intersection results:

$S_1 = \{T_{1,5}, T_3, T_4\}$

$S_2 = \{T_{2,6}, T_3, T_4\}$

$S_3 = \{T_{1,5}, T_3\}$

$S_4 = \{T_{1,5}, T_4\}$

$S_5 = \{T_3, T_4\}$

$S_6 = \{T_{2,6}, T_3\}$

$S_7 = \{T_{2,6}, T_4\}$

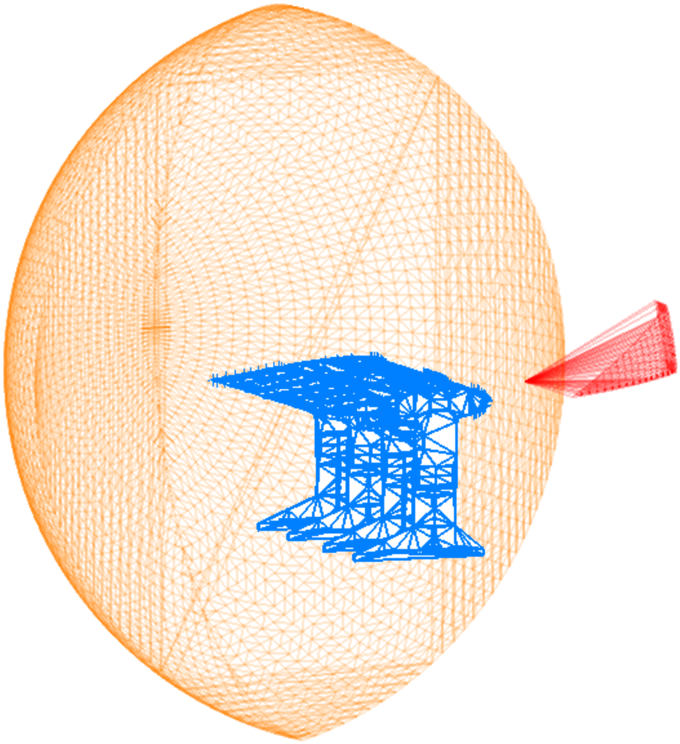
$S_8 = \{T_{1,5}\}$

$S_9 = \{T_{2,6}\}$

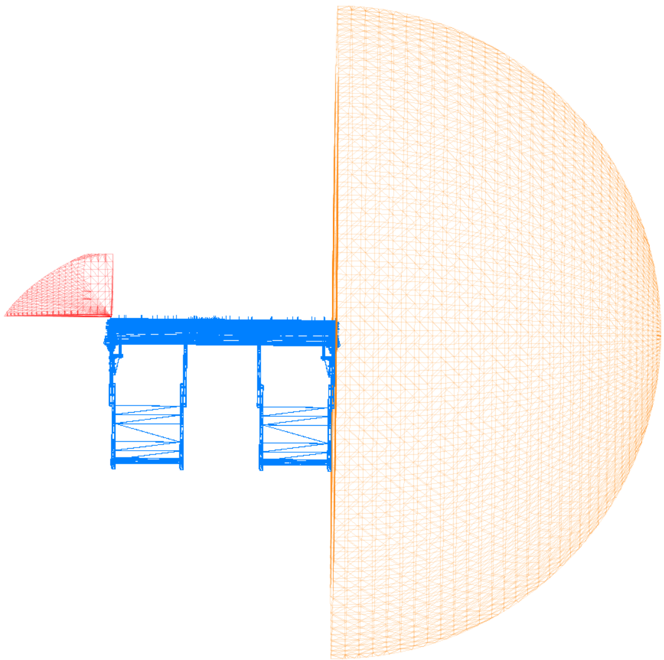
The corresponding binary matrix is then constructed as:

$$\begin{array}{c|ccccccccc|}
 & S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 \\
T_{1,5} & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
T_{2,6} & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
T_3 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
T_4 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0
\end{array}$$

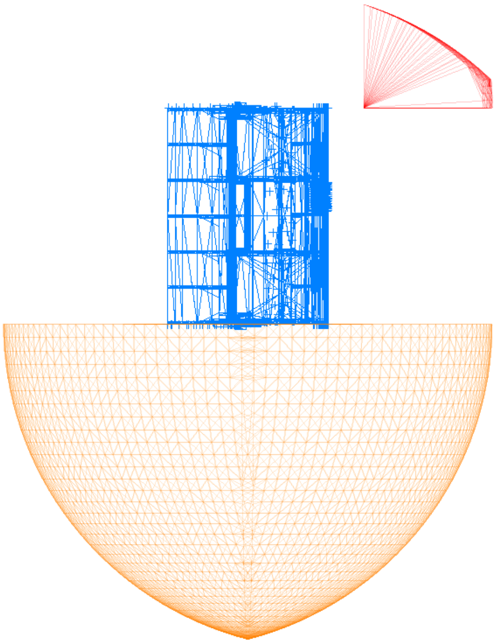
Note that there is more than one set cover in this case study using the greedy heuristic and$C = \{S_1, S_9\}$ is selected to cover all the inspection points. S_1 covers $T_{1,5}$, T_3 and T_4 while S_9 is used to locate the instrument to inspect $T_{2,6}$ with minimum uncertainty. Figure 9.19 gives the intersection results based on the clustering strategy. Other set covers can be used but are not described in this thesis.



(a) Oblique view of the clustered GICSs



(b) Front view of the clustered GICSs.



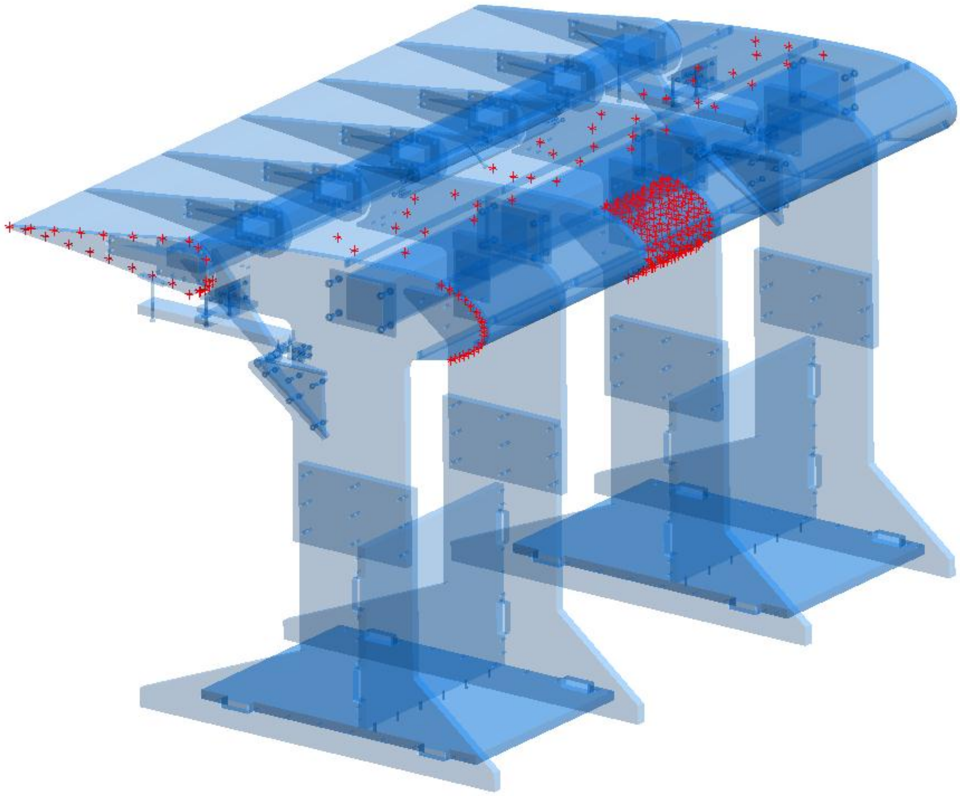
(c) Top view of the clustered GICSs

Figure 9.19 Clustering analysis results

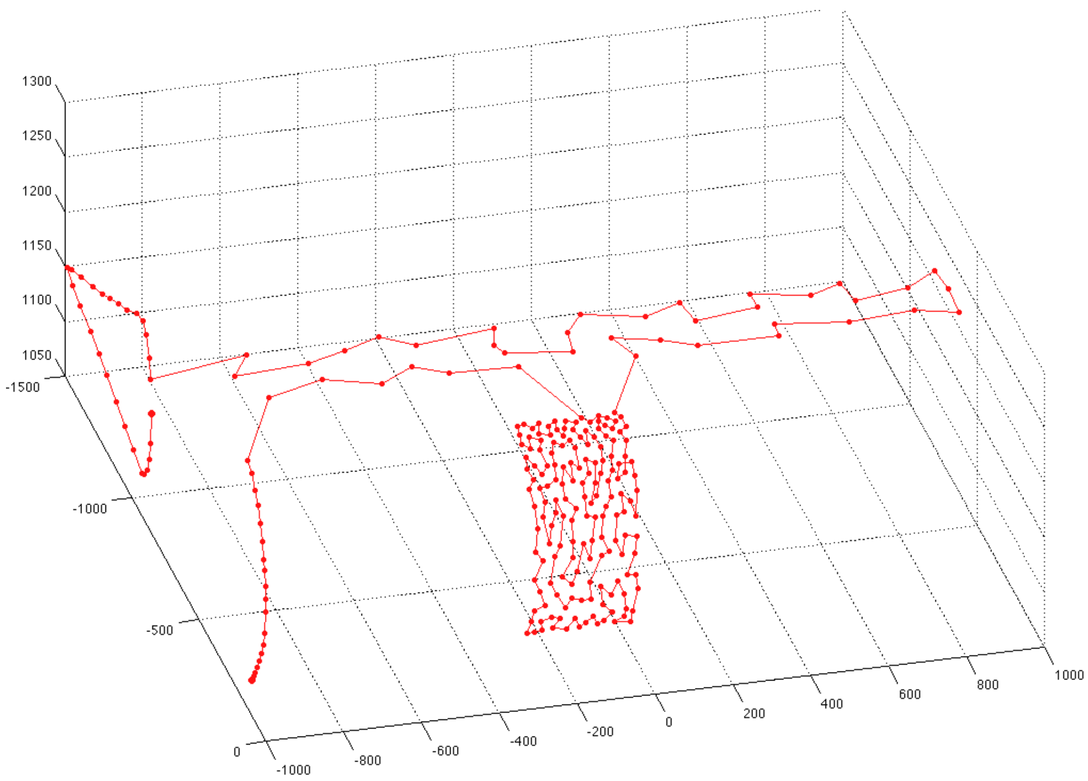
9.3.5 Measurement sequencing

As there are only two instrument locations existing in this particular example, the optimization of instrument locations is unnecessary. Measurement sequencing is conducted for the two clusters S_1 and S_9 respectively. 249 inspection points in S_1 are modelled as the TSP using 3D Euclidean distance (Figure 9.20(a)) and the proposed GA algorithm is carried out to generate the approximately optimal solution shown in Figure 9.20(b), which has the total length of 10943mm. The population size is set to 152 and the goal is achieved at 49744th iteration. The solution history is given in Figure 9.20(c).

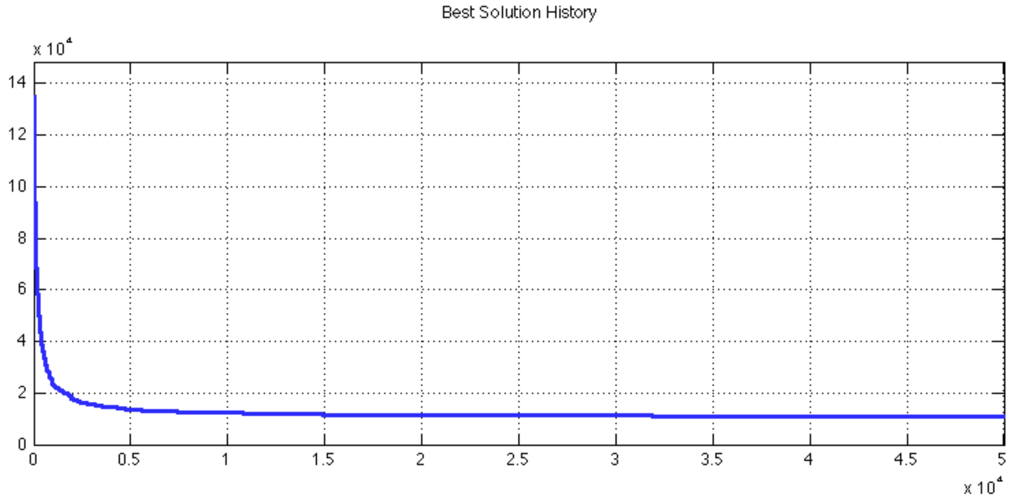
51 inspection points in S_9 are modelled as the TSP using 3D Euclidean distance (Figure 9.21(a)) and the proposed GA algorithm is carried out to generate the approximately optimal solution shown in Figure 9.21(b), which has the total length of 2403mm. The population size is set to 152 and the goal is achieved at 1473th iteration. The solution history is given in Figure 9.21(c).



(a) Distribution of inspection points in S_1

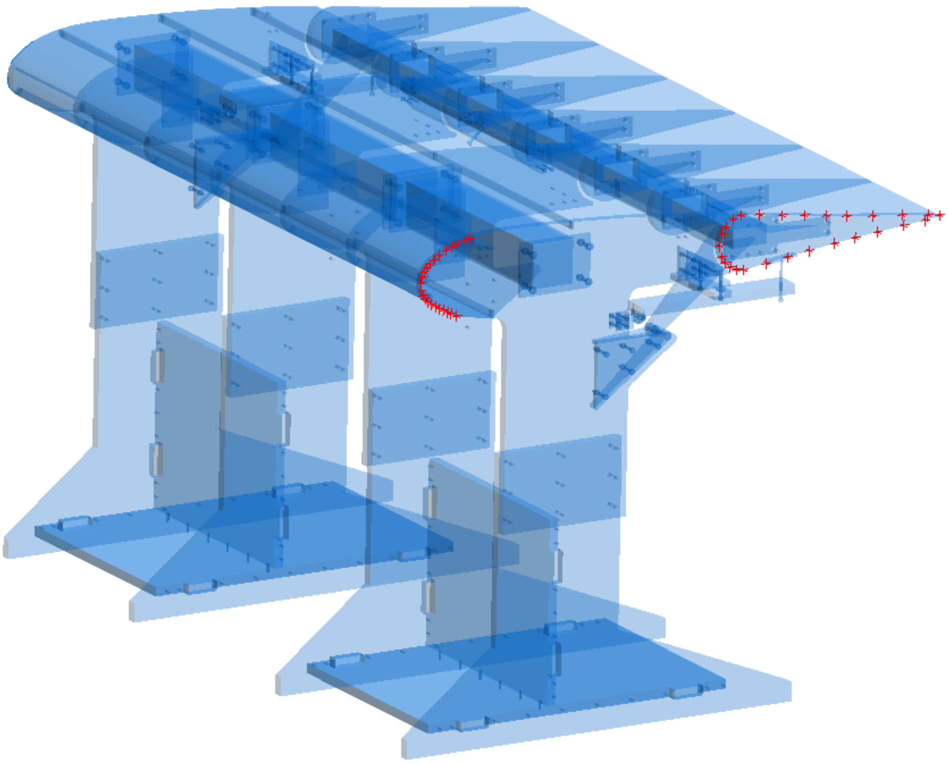


(b) Approximately optimal measuring sequence of inspection points in S_1

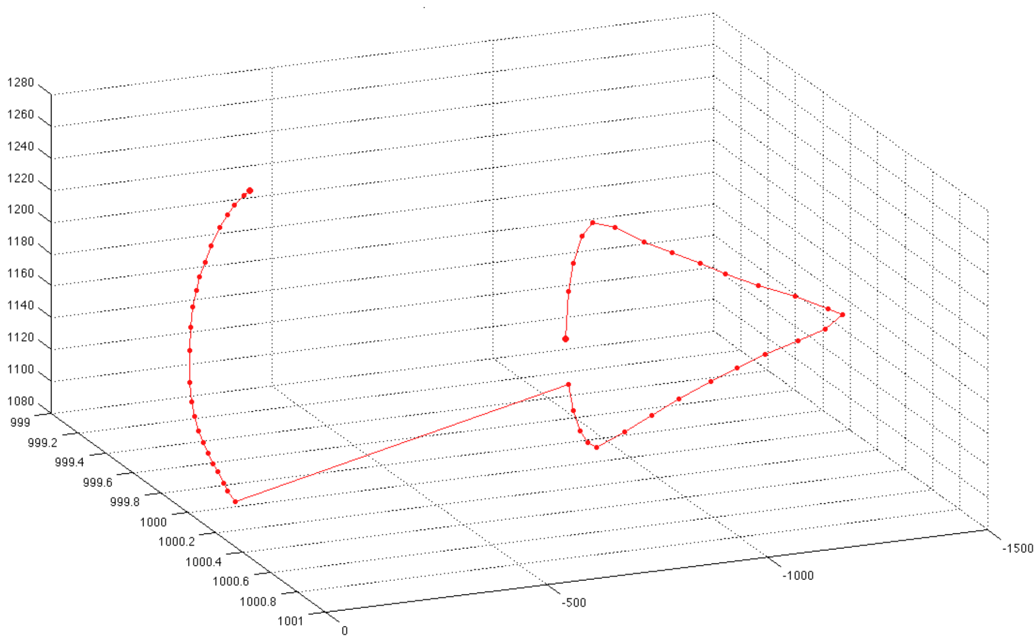


(c) Solution history

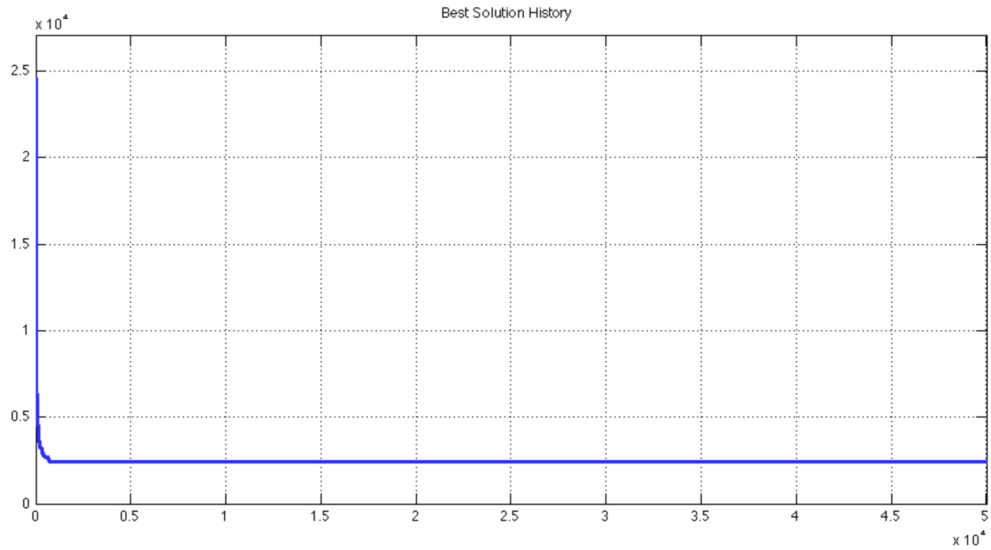
Figure 9.20 TSP modelling and solution of S_1



(a) Distribution of inspection points in S_9



(b) Approximately optimal measuring sequence of inspection points in S_9

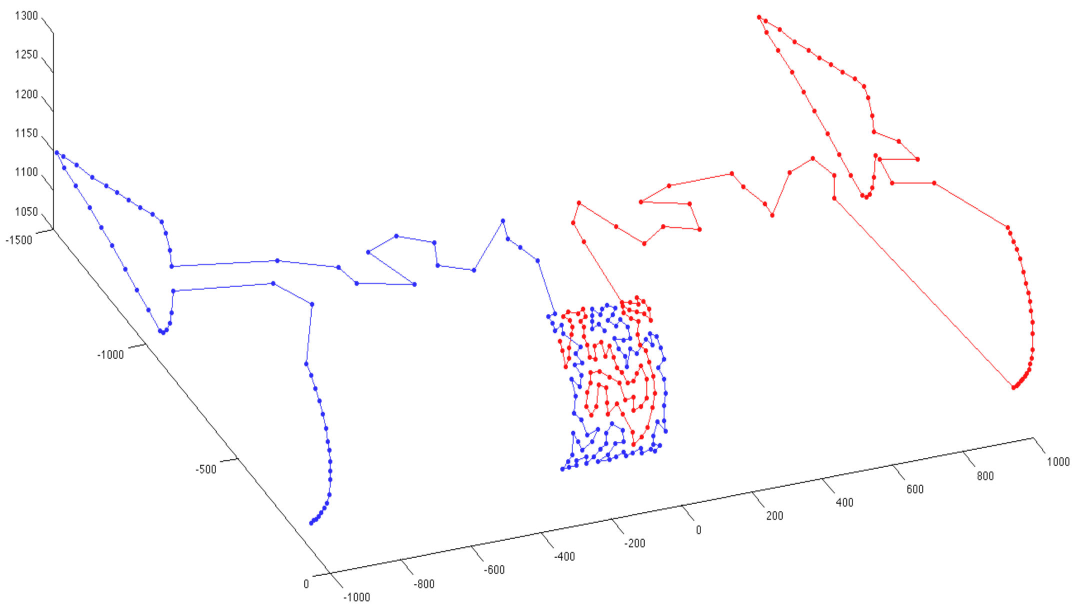


(c) Solution history

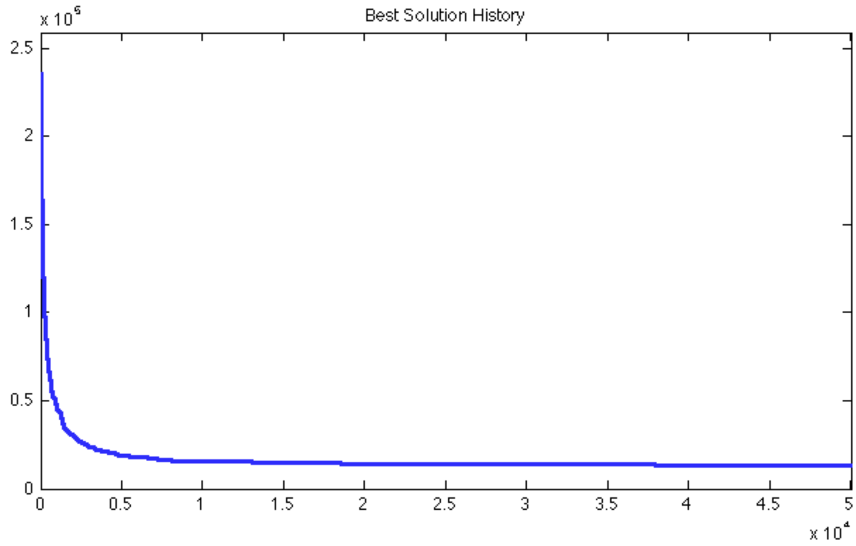
Figure 9.21 TSP modelling and solution of S_9

An attempt is also made to evaluate the framework in the case that multiple instruments are available using this example. Assuming two laser trackers can be deployed to accomplish the inspection task, the set cover is then selected as $C = \{S_1, S_2\}$ so that all the inspection points are covered. Additionally, T_3 and T_4 can be inspected by both laser trackers to maximise the efficiency. 300 inspection points in are modelled as the multiple TSP using 3D Euclidean distance and the proposed GA algorithm is carried out to generate the approximately optimal solution shown in Figure 9.22(a). The population size is set to 120 and the goal is achieved at 49071th iteration. The solution history is given in Figure 9.22(b).

The total length of this path is 13329mm, which is just 20mm shorter than the sum of the two previous individual paths. However, the inspection time can be shortened by half as each laser tracker is used to inspect only 150 points and two laser tracker work simultaneously without changing the location of the station. It is also beneficial for the production environment to simplify the instrument setup and configuration process, which results in a shorter lead time.



(a) Approximately optimal measuring sequence of inspection points for two laser trackers



(b) Solution history

Figure 9.22 Multiple TSP modelling and solution for two laser trackers

9.4 Summary

In this chapter, the proposed inspection planning framework is demonstrated by two case studies. A large-scale artefact that includes common features such as plane, hole, fillet and pocket is presented firstly to evaluate the work flow of the framework. Typical GD&T is assigned with the part including position, flatness, profile, parallelism, and dimension as well as a datum system. Inspection tasks are identified based on the extracted GD&T information, which lays the foundation for selecting the most appropriate instrument, namely laser tracker in this case. Global instrument configuration space is generated for both convex and concave regions on the artefact and then aggregated according to the results of clustering analysis. Laser tracker is located optimally where the minimum sum of uncertainties to all inspection points is achieved and finally, the approximately optimal measuring sequence is generated using the developed GA to guide the metrologists to conduct the inspection.

The second case study involves an assembly wing box, which is a typical aerospace industrial part. Inspection points and distribution are given with the assignment and the framework successfully extracts the relevant information for the instrument selection process. The remaining processes are effectively demonstrated on the wing box and an approximately optimal measuring sequence is outputted in the end. In addition, an attempt is made to explore the performance of the system under the scenario where multiple instruments are available. Measurement sequences for two instruments are obtained using multiple TSP modellers. It is observed that inspection time can be reduced significantly by carrying out two sets of inspection tasks simultaneously and production lead time can benefit from the simplified instrument setup and configuration process.

CHAPTER 10 CONCLUSION AND FUTURE WORK

10.1 Conclusion and contributions

A systematic GD&T based inspection planning framework is proposed in this study for large volume metrology (LVM) instruments. Currently, rapidly advancing LVM technologies strongly demand a formally modelled LVM inspection system, which is capable of participating in the design and inspection stage within large-scale manufacturing activities. However, this is absent from the literature and inspection planning methods for CMMs cannot be directly adopted due to the inherent differences of measuring principle. With the purpose of solving problems from what to measure to how to measure, the proposed comprehensive system contains seven functions that support: tolerance feature analysis, instrument selection, inspection point selection, accessibility and visibility analysis, instrument setup and configuration, clustering analysis and measurement sequencing. The novel methodologies defined and utilized in this work coupled with contributions, can be summarised as follows:

a) A new taxonomy of GD&T is proposed to decompose the designed tolerancing information and obtain basic inspection tasks efficiently. The integration of this process with metrology software eliminates the compatibility issue due to the variety of model formats adopted by different modelling software.

b) Intuitionistic fuzzy decision making method is implemented to facilitate the instrument selection process in early product development stages where vagueness exists in weighting the criteria and rating the performance. Measurability characteristics are first identified with respect to specific inspection task and grouped into quantitative (crisp) and qualitative (fuzzy) attributes. An instrument filtration procedure eliminates the unsuitable instruments according to the crisp MCs and the remaining instruments are further investigated based on fuzzy criteria, which are assessed by linguistic importance and performance rating. A rank list consisted of capable instruments is eventually output to assist instrument selection by the user.

c) Visibility analysis for LVM instruments is conducted with the assistance of CAD rendering techniques to solve the question of where to locate the instrument. The definition of visibility cone (VC) is extended using the measurable space of instrument instead of unit sphere. By applying computational graphics approaches such as visibility culling techniques and bounding volume, the global instrument configuration space is generated. This innovative implementation saves enormous computation power and realizes the complex intersection of 3D visibility cones (VC) in clustering process.

d) Having obtained the viewable space of all inspection points, a clustering process is carried out to minimize the relocation of the instrument. Due to the absence of methods that can find the appropriate cluster that cover all the 3D continuous configuration spaces, set cover problem is innovatively selected to model the clustering process and a greedy heuristic is employed to find the approximately optimal solution. All inspection points are covered by the generated clusters while maximum coverage of each cluster is achieved.

e) Instrument configuration takes place after the clustering process with the aim of minimizing the combined uncertainty with respect to all inspection points within each cluster. An exact solution can be obtained utilizing the optimization tool provided by Matlab on the condition that the uncertainty model of the instrument is available. Alternatively, an approximate solution is generated by discretising the solution space and simulating the measurements.

f) Inspection path planning is modelled as single/multiple 3D travelling salesperson problem (TSP) coupled with the application of geodesic distance, which ensures that the real optimum path is identified regardless of the geometry of the part and distribution of inspection points. A genetic algorithm is developed to solve the modelled TSP within a satisfactory approximation factor. Inspection time is reduced significantly with the optimal path and better results are achieved when multiple instruments are engaged following the optimum paths respectively.

g) Although the system closes the loop as a complete function in design-plan-inspection, each of the modules are able to serve the inspection activity individually through the interface to metrology software e.g. visibility analysis can be implemented as a line of sight check for any user defined target-instrument setup and the measurement sequencing module is able to provide the optimal inspection sequence to user-inputted inspection assignment.

10.2 Limitations and future work

The methodology presented and the system developed in the current study, however, has limitations. Since this research is the first attempt to develop an inspection planning system for LVM instruments, various potential opportunities of refining and enhancing the functionalities of the prototype are expected in the future. The author is aware of certain limitations and therefore summarizes the corresponding future research as follows:

a) Fully automating the GD&T extraction and analysis process.

Although the underpinned taxonomy is proposed to decompose the GD&T information into inspection task, there is still a need to fully automate this process. Currently only the extraction of GD&T information is automatic but the analysis process requires human interaction. It requires an efficient approach to handle the huge amount of data when vast GD&T information is involved. One way to tackle the above problems is to integrate the analysis process into metrology software e.g. Spatial Analyzer, where design information can be obtained and the effectiveness of this process is maximized and appreciated by the user. It is worth pursuing more details of this novel methodology since it is the first one decomposing design information into basic measurement elements e.g. point, line and surface, which lays the ground for the subsequent activities.

b) Improving the instrument selection algorithm.

The instrument filtration can be refined with more clearly defined measurability characteristics. The user may add supplementary requirements to accommodate the specific task. Additionally, research regarding fuzzy multi-attributes decision-making (MADM) is rapidly advancing nowadays. It might be possible that better MADM methods will emerge or the current adopted methodology can be developed to fulfil the specific metrology-decision requirement. It is preferable to integrate the instrument selection process within metrology software although currently the process can be automated in spread sheet.

c) Establishing new standards and best practice rules for identifying the number of inspection points and corresponding distribution.

At present the process follows the guidance and rules in BSI and ASME standards. Having had a variety of new metrology technologies, new methodologies are urgently demanded to determine the number of inspection points and the approach of distribution, with respect to specific tolerance and confidence requirements. Nevertheless, research in this area is out-of-date and exclusively for CMMs, according to the literature. A thorough review can be composed coupled with identification of latest research requirements. New standards and best practice guide for LVM can be established with appropriate institutions, which will be well accepted and applied. Again, automating the process within metrology software is preferable.

d) Enhancing the visibility analysis and instrument localization.

The visibility analysis is carried out for discrete point only at the moment. Future research concerning feature visibility is required, which increases the efficiency of the intersection process when the number of inspection points is considerable. Integration with metrology software may be more effective than with CAD software since the results are for the following inspection activities, whereas this is mainly limited by the rendering methodology used in corresponding software. Alternatively, a fundamental study with respect to visibility and its application using both computational graphic e.g. aspect graph, ray casting and hardware e.g. z-buffer can be conducted.

e) Improving the uncertainty models of LVM instruments.

Improved uncertainty model of laser tracker is able to contribute to more accurate instrument configuration and corresponding minimization approach demands further investigation. Other instruments are located using measurement simulation at present due to the lack of uncertainty model. Future research can focus on the incorporation of appropriate mathematical models of other LVM instruments and implementation of the uncertainty simulation process.

f) Applying measurement sequence planning for more instruments.

The current sequence planning is only valid for point-based measurement. Modification is necessary if scanning instrument is considered. Moreover, the process can be integrated with robot path planning to realize automated inspection e.g. industrial robot and platform. As far as software integration is concerned, current metrology software coupled with robot control workbench match the proposed method perfectly.

REFERENCES

Ajmal, A. and Zhang, S. G., 1998. The application of a knowledge-based clustering algorithm as an aid to probe selection and inspection process planning. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 212(4), pp. 299-305.

Al-Dulaimi, B. and Ali, H., 2008. Enhanced traveling salesman problem solving by genetic algorithm technique (TSPGA). *Proceeding of the world Academy of Science, Engineering and Technology*, Rome, Italy, pp. 296-302.

Albuquerque, V. A., Liou, F. W. and Mitchell, O. R., 2000. Inspection point placement and path planning algorithms for automatic CMM inspection. *International Journal of Computer Integrated Manufacturing*, 13(2), pp. 107-120.

Aleksandrov, L., Djidjev, H., Guo, H., Maheshwari, A., Nussbaum, D. and Sack, J. R., 2006. Approximate shortest path queries on weighted polyhedral surfaces. *Mathematical Foundations of Computer Science*, 4162, pp. 98-109.

Alting, L. and Zhang, H., 1989. Computer aided process planning: the state-of-the-art survey *International Journal of Production Research*, 27(4), pp. 553-585.

Alvarez, B. J., Fernandez, P., Rico, J. C., Mateos, S. and Suarez, C. M., 2008a. Accessibility analysis for automatic inspection in CMMs by using bounding volume hierarchies. *International Journal of Production Research*, 46(20), pp. 5797-5826.

Álvarez, B. J., Fernández, P., Rico, J. C. and Valiño, G., 2008b. Accessibility Analysis for the Automatic Contact and Non-contact Inspection on Coordinate Measuring Machine. *Proceedings of the World Congress on Engineering*, London, U.K.

Applegate, D. L., Bixby, R. E., Chv´atal, V. and Cook, W. J., 2003. Implementing the Dantzig–Fulkerson–Johnson algorithm for large scale traveling salesman problems. *Math Program Ser B*, 97, pp. 91-153.

Applegate, D. L., Bixby, R. E., Chv´atal, V. and Cook, W. J., 2006. The traveling salesman problem: a computational study, Princeton University Press.

Applegate, D. L., Bixby, R. E., Chv´atal, V., Cook, W. J. E., D.G., Goycoolea, M. and Helsgaun, K., 2009. Certification of an optimal TSP tour through 85900 cities. *Operations Research Letters*, 37(1), pp. 11-15.

ASME Y14.5, 2009. Dimensioning and Tolerancing. ASME.

Atanassov, K. T., 1986. Intuitionistic fuzzy sets. *Fuzzy Sets and Systems*, 20, pp. 87-96.

B89.4.19, 2006. Performance evaluation of laser-based spherical Coordinate Measurement System. American Society of Mechanical Engineers.

B89.7.3.1, 2001. Guidelines for decision rules: considering measurement uncertainty in determining conformance to specifications. American Society of Mechanical Engineers.

Balasubramaniam, M., Sarma, S. E. and Marciniak, K., 2003. Collision-free finishing toolpaths from visibility data. *Computer-Aided Design*, 35(4), pp. 359-374.

Barreiro, J., Labarga, J. E., Vizan, A. and Rios, J., 2003. Functional model for the development of an inspection integration framework. *International Journal of Machine Tools Manufacture*, 43(15), pp. 1621-1632.

Basch, J., Guibas, L., Ramkumar, G. and Ramshaw, L., 1996. Polyhedral tracings and their convolution. *In: Proceedings of the 2nd workshop on algorithmic foundations of robotics.*, pp. 171-195.

Beasley, J. E., 1987. An algorithm for set covering problems. *European Journal of Operational Research*, 31, pp. 85-93.

Beasley, J. E., 1990. A Lagrangian heuristic for set-covering problems. *Naval Research Logistics*, 37, pp. 151-164.

Beasley, J. E. and Chu, P. C., 1996. A genetic algorithm for the set covering problem. *European Journal of Operational Research*, 94, pp. 392-404.

Beasley, J. E. and Jornsten, K., 1992. Enhancing an algorithm for set covering problem. *European Journal of Operational Research*, 58, pp. 293-300.

Beg, J. and Shunmugam, M. S., 2002. An Object Oriented Planner for Inspection of Prismatic Parts – OOPIPP. *The International Journal of Advanced Manufacturing Technology*, 19(12), pp. 905-916.

Biggs, N. L., Lloyd, E. K. and Wilson, R. J., 1986. Graph Theory 1736-1936. Clarendon Press, Oxford.

Boran, F. E., Genc, S., Kurt, M. and Akay, D., 2009. A multi-criteria intuitionistic fuzzy group decision making for supplier selection with topsis method. *Expert System Application*, 36(8), pp. 11363-11368.

Boukreev, K., 2001. Genetic algorithms and the traveling salesman problem [online]. Available from: http://www.codeproject.com/articles/1403/genetic-algorithms-and-the-traveling-salesman-prob [Accessed 12 April 2011]

Bozdağ, C. E., Kahraman, C. and Ruan, D., 2003. Fuzzy group decision making for selection among computer integrated manufacturing systems. *Computers in Industry*, 51, pp. 13-29.

Brown, C. W. and Gyorog, D. A., 1990. Generative inspection process planner for integrated production. *American Society of Mechanical Engineers, Production Engineering Division* 47, pp. 151-162.

Brummit, B. and Stentz, A., 1996. Dynamic mission planning for multiple mobile robots. *Proceedings of the IEEE international conference on robotics and automation.*

Brummit, B. and Stentz, A., 1998. GRAMMPS: a generalized mission planner for multiple mobile robots. *Proceedings of the IEEE international conference on robotics and automation*.

Brusco, M. J., Jacobs, L. W. and Thompson, G. M., 1999. A morphing procedure to supplement a simulated annealing heuristic for cost- andcoverage-correlated set-covering problems. *Annals of Operations Research*, 86(0), pp. 611-627.

Bryman, A., 2004. Social resarch methods, Oxford University Press.

BS EN ISO 1101, 2005. Geometrical Product Specifications (GPS)-Geometrical Tolerancing-Tolerances of Form, Orientation, Location and Run-out. British Standard Institute.

BS 7172, 1989. Guide to assessment of position, size and departure from nominal form of geometric features. British Standard Institute.

BS EN ISO 14253-1, 1999. Geometrical product specifications (gps) —inspection by measurement of workpieces and measuring equipment—part 1: decision rules for proving conformance or non-conformance with specifications. British Standard Institute.

BS ISO/TR 14638, 1995. Geometrical product specification (GPS) - Masterplan. British Standard Institute.

BS ISO/IEC 19793, 2008. Information technology - open distributed processing: use of UML for ODP viewpoint specifications. British Standard Institute.

Buckles, B. P., Petry, P. E. and Kuester, R. I., 1990. Schema survival rates and heuristic search in genetic algorithms. *Proceedings of the 2nd International IEEE Conference on Tools for Artificial Intelligence,* Herndon, VA ,USA, pp. 322-327.

Cai, B., Dai, W., Jamshidi, J. and Maropoulos, P. G., 2010. Measurability characteristics mapping for large volume metrology instruments selection. *Proceedings of the Advances in Manufacturing Technology - XXIII, 7th International Conference on Manufacturing Research*, Warwick, UK, pp. 438-442.

Cai, B., Guo, Y., Jamshidi, J. and Maropoulos, P. G., 2008. Measurability analysis of large volume metrology process model for early design. *Proceedings of 5th*

International Conference on Digital Enterprise Technology, Nants, France, pp. 807-820.

CGAL, Computational Geometry Algorithms Library (CGAL) [online]. Available from: http://www.cgal.org/ [Accessed 10 November 2011]

Chan, F. T. S., N., K., Tiwari, M. K., Lau, H. C. W. and Choy, K. L., 2008. Global supplier selection: a fuzzy-ahp approach. *International Journal of Production Research*, 46(14), pp. 3825-3857.

Chen, C. T., Lin, C. T. and Huang, S. F., 2006. A fuzzy approach for supplier evaluation and selection in supply chain management. *International Journal of Production Economics*, 102, pp. 289-301.

Chen, J. and Han, Y., 1996. Shortest paths on a polyhedron Part I: computing shortest paths. *International Journal of Computational Geometry and Applications*, 6(2), pp. 127-144.

Chen, L. L., Chou, S. Y. and Woo, T. C., 1993. Parting directions for mould and die design. *Computer-Aided Design*, 25(12), pp. 762-768.

Chen, L. L., Chou, S. Y. and Woo, T. C., 1993. Separating and Intersecting Spherical Polygons: Computing Machinability on Three-, Four-, and Five-axis Numerically Controlled Machines. *ACM Transactions on Graphics*, 12(4).

Chen, L. L. and Woo, T. C., 1992. Computational geometry on the sphere with application to automated machining. *Journal of Mechanical Design*, 114(2), pp. 288-295.

Cheng, C., Lee, W. and Wong, K., 2002. A genetic algorithm-based clustering approach for database partitioning. *IEEE Transactions on Systems, Man and Cybernetics*, 32(3), pp. 215-230.

Chin, K. S., Zheng, L. Y. and L.Wei, 2003. A hybrid rough-cut process planning for quality. *International Journal of Advanced Manufacturing Technology*, 22(9), pp. 733-743.

Cho, M. W. and Kim, K., 1995. New inspection planning strategy for sculptured surfaces using coordinate measuring machine. *International Journal of Production Research*, 33(22), pp. 427-444.

Cho, M. W., Lee, H., Yoon, G. S. and Choi, J., 2004. A computer-aided inspection planning system for on-machine measurement-Part II: Local inspection planning. *KSME International Journal*, 18, pp. 1358-1367.

Cho, M. W., Lee, H., Yoon, G. S. and Choi, J., 2005. A feature-based inspection planning system for coordinate measuring machines. *International Journal of Advanced Manufacturing Technology*, 26, pp. 1078-1087.

Cho, M. W. and Seo, T. L., 2002. Inspection planning strategy for the on-machine measurement process based on CAD/CAM/CAI integration. *International Journal of Manufacturing Technology*, 19, pp. 607-617.

Cho, Y. H., 2010. An efficient solving the travelling salesman problem: global optimization of neural networks by using hybrid method. Traveling salesman problem, theory and applications. D. Davendra, InTech.

Chvatal, V., 1979. A Greedy Heuristic for the Set-Covering Problem. *Mathmatics of Operations Research*, 4, pp. 233-235.

Cristofolini, I., Concheri, G., Meneghello, R. and Wolf, G., 2001. Geometric dimensioning and tolerancing (GD&T) versus geometrical product specification (GPS). *In Proceeding of XII ADM International Conference*, Rimini, Italy, pp. 38-51.

Current, J. and O'Kelly, M., 1992. Locating Emergency Warning Sirens. *Decision Sciences,* 23, pp. 221-234.

Cuypers, W., VanGestel, N., Voet, A., Kruth, J. P., Mingneau, J. and Bleys, P., 2009. Optical measurement techniques for mobile and large-scale dimensional metrology. *Optics and Lasers in Engineering*, 47, pp. 292-300.

Dai, W. and Tang, X. Q., 2008. Quality plan model for product development. *Proceedings of the 38th International Conference on Computers and Industrial Engineering*, Beijing, China, 2, pp. 1535-1541.

Dantzig, G. B., Fulkerson, D. R. and Johnson, S. M., 1954. Solution of a large-scale traveling salesman problem. *Operations Research*, 2, pp. 393-410.

DassaultSystèmes, 2000. The Surfaces of CATIA Geometric Modeler [online]. Available from: http://www.maruf.ca/files/caadoc/caagobtecharticles/surfaces.htm [Accessed 05 January 2012]

Davis, L., 1985. Job shop scheduling with genetic algorithms. *Proceedings of 1st International Conference on Genetic Algorithms,* pp. 136-140.

Department of Defence, 2006. Defence acquisition guidebook. USA.

Desrosiers, J., Dumas, Y., Solomon, M. M. and Soumis, F., 1995. Chapter 2 Time constrained routing and scheduling. *Handbooks in Operations Research and Management Science*. M.O. Ball, T. L. Magnanti, C. L. Monma and G. L. Nemhauser, Elsevier. Volume 8: 35-139.

Dhaliwal, S., Gupta, S. K. and Priyadarshi, J. H. A., 2003. Algorithms for computing global accessibility cones. *Journal of Computing and Information Science in Engineering*, 3(3), pp. 200-209.

Ding, S., Mannan, M. A. and Poo, A. N., 2004. Oriented bounding box and octree based global interference detection in 5-axis machining of free-form surfaces. *Computer-Aided Design*, 36(13), pp. 1281-1294.

Durand, F., 2000. A multidisciplinary survey of visibility. Extract of the PhD dissertation.

EIMaraghy, H., 1993. Evolution and future perspectives of CAPP. *Annals of the CIRP*, 42, pp. 1-23.

EIMaraghy, H. and Gu, P., 1987. Expert system for inspection planning. *Annals of the CIRP*, 36(1), pp. 85-89.

ElMaraghy, H. and Yang, X., 2003. Computer-aided planning of laser scanning of complex geometries. *Annals of the CIRP*, 52(1), pp. 411-414.

Estler, W. T., Edmundson, K. L., N., P. G. and Parker, D. H., 2002. Large-scale metrology - an update. *Annals of the CIRP*, 51(2), pp. 587-609.

Fan, K. C. and Leu, M. C., 1998. Intelligent planning of CAD-directed inspection for coordinate measuring machines. *Computer Integrated Manufacturing Systems*, 11(1), pp. 43-51.

Feng, S., 1994. Dimensional inspection planning based on product data standards. *Proceedings of Concurrent Engineering: Research and Applications Conference and Application*, pp. 333-342.

Fernández, P., Rico, J. C., Álvarez, B. J., Valiño, G. and Mateos, S., 2008. Laser scan planning based on visibility analysis and space partitioning techniques. *The International Journal of Advanced Manufacturing Technology*, 39(7), pp. 699-715.

Fisher, M. L. and P. Kedia, 1990. Optimal solution of set covering/partitioning problems using dual heuristics. *Management Science*, 36, pp. 674-688.

Flack, D. and Hannaford, J., 2005. *Measurement good practice guide No.80 fundamental good practice in dimensional metrology*. London, National Physical Laboratory.

Fogel, E. and Halperin, D., 2007. Exact and efficient construction of Minkowski sums of convex polyhedra with applications. *Computer-Aided Design*, 39(11), pp. 929-940.

Gao, J., Gindy, N. and Chen, X., 2006. An automated GD&T inspection system based on non-contact 3D digitization. *International Journal of Production Research*, 44(1), pp. 117-134.

Gao, J., Zheng, D. T., Gindy, N. and Clark, D., 2004. Extraction/conversion of geometric dimensions and tolerances for machining features. *The International Journal of Advanced Manufacturing Technology*, 26(4), pp. 405-414.

Garey, M. R. and Johnson, D. S., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness. San Francisco, W.H. Freeman.

Ge, Q., Chen, B., Smith, P. and Menq, C. H., 1992. Tolerance specification and comparative analysis for computer-integrated dimensional inspection. *International Journal of Production Research*, 30(9), pp. 2173-2197.

Goldberg, D. E., 1989. Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley,.

Grossman, T. and Wool, A., 1994. Computational experience with approximation algorithms for the set covering problem, Theoretical Division and CNLS, Los Alamos National Laboratory, Los Alamos.

Grötschel, M., Jünger, M. and Reinelt, G., 1991. Optimal Control of Plotting and Drilling Machines: A Case Study. *Mathematical Methods of Operations Research*, 35(1), pp. 61-84.

Guibas, L. and Seidel, R., 1986. Computing convolutions by reciprocal search. *In: SCG '86: Proceedings of the second annual symposium on computational geometry.*, pp. 90-99.

Haouari, M. and Chaouachi, J. S., 2002. A probabilistic greedy search algorithm for combinatorial optimisation with application to the set covering problem. *Journal of the Operational Research Society*, 53(7), pp. 792-799.

Harche, F. and Thompson, G. L., 1994. The column subtraction algorithm: An exact method for solving weighted set covering, packing and partitioning problems. *Computers and Operations Research*, 21, pp. 689-705.

Helmy, H. A., 1991. Feature recognition and CAD-directed inspection using solid geometric representation. PhD Thesis, Lehigh University.

Hert, S. and Schirra, S., 3D Convex Hulls [online]. Available from: http://www.cgal.org/manual/latest/doc_html/cgal_manual/convex_hull_3/chapter_main.html [Accessed 10 November 2011]

Hoffman, K. L. and Padberg, M., 1993. Solving airline crew scheduling problems by branch-and-cut. *Management Science*, 39(6), pp. 657-682.

Hopp, T. H., 1984. CAD-directed inspection. *Annals of the CIRP*, 33(1), pp. 357-361.

Hopp, T. H. and Lau, K. C., 1983. A hierarchical model-based control system for inspection. *Proceedings of the 5th International Symposium on Automated Integrated Manufacturing*, pp. 169-187.

Huo, D. H., Maropoulos, P. G. and Cheng, C. H., 2010. The framework of the virtual laser tracker: a systematic approach to the assessment of error sources and uncertainty in laser tracker measurement. *Proceedings of the 6th CIRP-Sponsored International Conference on Digital Enterprise Technology*, pp. 507-523.

Hwang, C. Y., Tsai, C. Y. and Chang, C. A., 2004. Efficient inspection planning for coordinate measuring machines. *International Journal of Computer Integrated Manufacturing*, 23, pp. 732-742.

Hwang, I., Lee, H. and Ha, S., 2002. Hybrid neuro-fuzzy approach to the generation of measuring points for knowledge-based inspection planning. *International Journal of Production Research*, 40(11), pp. 2507-2520.

ISO/IEC 19505-1, 2012, Information technology - object management group unified modelling language (OMG UML) - Part 1, International Standard Organisation.

ISO/IEC 19505-2, 2012, Information technology - object management group unified modelling language (OMG UML) - Part 2, International Standard Organisation.

Ilushin, O., Elber, G., Halperin, D., Wein, R. and Kim, M.-S., 2005. Precise global collision detection in multi-axis NC-machining. *Computer-Aided Design*, 37(9), pp. 909-920.

Innovmetric, 2012. PolyWorks [online]. Available from: http://www.innovmetric.com/polyworks/3d-scanners/so_dimensional2.aspx?lang=en [Accessed 15 March 2012]

Itoh, H., 2010. The method of solving for travelling salesman problem using genetic algorithm with immune adjustment mechanism. Traveling salesman problem, theory and applications. D. Davendra, InTech.

Jackman, J. and Parka, D.-K., 1998. Probe orientation for coordinate measuring machine systems using design models. *Robotics and Computer-Integrated Manufacturing*, 14(3), pp. 229-236.

Jeffries, C., 1991. Code recognition and set selection with neural networks. Boston, Birkhauser.

Johnson, D. S., 1974. Approximation algorithms for combinatorial problem. *Jounal of Computer System*, 9, pp. 256-278.

Johnson, G. and Johnson, P., 1991. Resarch methods for managers, Sage.

Joshi, S. and Chang, T. C., 1988. Expert process planning system with solid model interface. *International Journal of Production Research*, 26(5), pp. 864-885.

Joshi, S. and Chang, T. C., 1988. Graph-based Heuristics for recognition of machined features from a 3D Solid model. *Computer-Aided Design*, 20(2), pp. 58-66.

Kanai, T. and Suzuki, H., 2001. Approximate shortest path on a polyhedral surface and its applications. *Computer-Aided Design*, 33(11), pp. 801-811.

Karsak, E. E., 2002. Distance-based fuzzy MCDM approach for evaluating flexible manufacturing system alternatives. *International Journal of Production Research*, 40(13), pp. 3167-3181.

Khardekar, R., Burton, G. and McMains, S., 2006. Finding feasible mold parting directions using graphics hardware. *Computer-Aided Design*, 38(9), pp. 327-341.

Kim, D. S., 1990. Cones on bezier curves and surfaces. Ann Arbor, MI, USA, PhD Thesis.

Kim, D. S., Papalambros, P. Y. and Woo, T. C., 1995. Tangent, normal, and visibility cones on bezier surfaces. *Computer Aided Geometric Design*, 12(3), pp. 305-320.

Kimmel, R. and Sethian, J., 1998. Computing geodesic paths on manifolds. *National Academy of Sciences of the USA*, 95, pp. 8431-8435.

Kimmerl, S., 2005. Real-Time Collision Detection for Dynamic Virtual Environment [online]. University of Tübinge. Available from: http://www-ljk.imag.fr/publications/basilic/com.lmc.publi.publi_inproceedings@117681e94b6_1860ffd/bounding_volume_hierarchies.pdf [Accessed 15 November 2011]

Klosowski, J. T., Held, M., Mitchell, J. S. B., Sowizral, H. and Zikan, K., 1998. Efficient collision detection using bounding volume hierarchies of k-DOPs. *IEEE Transactions on Visualization and Computer Graphics*, 4(1), pp. 21-36.

Kolen, A. and Pesch, E., 1994. Genetic local search in combinatorial optimization. *Discrete Applied Mathematics*, 48, pp. 273-284.

Kugel, H. W., Loesser, D., Roquemore, A. L., Menon, M. M. and Barry, R. E., 2001. Precision metrology of nstx surfaces using coherent laser radar ranging. *Review of Scientific Instruments*, 72(1), pp. 533-536.

Kurekova, E., 2001. Measurement process capability – trend and approaches. *Measurement Science Review*, 1(1), pp. 43-46.

Kweona, S. and Medeiros, D. J., 1998. Part orientations for CMM inspection using dimensioned visibility maps. *Computer-Aided Design*, 90(9), pp. 741-749.

LaLena, M., 2010. Traveling salesman problem using genetic algorithms [online]. Available from: http://www.lalena.com/ai/tsp/ [Accessed 14 April 2011]

Laporte, G., 1992. The traveling salesman problem: an overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(2), pp. 231-247.

Lau, K., Hocken, R. and Haight, W., 1986. Automatic laser tracking interferometer system for robot metrolog. *Precision Engineering*, 8(1), pp. 3-8.

Lawler, E. L., Klenstra, J., Rinnooy, A. J. K. and Shmoys, D. B., 1985. The travelling salesman problem, John Wiley.

Lee, G., Mou, J. and Shen, Y., 1997. Sampling strategy design for dimensional measurement of geometric features using coordinate measuring machine. *International Journal of Machining Tools and Manufacturing*, 37(7), pp. 917-934.

Lenstra, J. K. and Rinnooy Kan, A. H. G., 1974. Some simple applications of the traveling salesman problem. *Stichting Mathematisch Centrum*, 38, Amsterdam.

Lenstra, J. K. and Rinnooy Kan, A. H. G., 1975. Some simple applications of the traveling salesman problem. *Operational Research Quarterly*, 26, pp. 717-733.

Li, Y. and Frank, M. C., 2007. Computing non-visibility of convex polygonal facets on the surface of a polyhedral cad model. *Computer-Aided Design*, 39(9), pp. 732-744.

Li, Y. and Gu, P., 2004. Free-form surface inspection techniques state of the art review. *Computer-Aided Design*, 36, pp. 1395-1417.

Lim, C. P. and Menq, C. H., 1994. CMM Feature accessibility and path generation. *International Journal of Production Research*, 32, pp. 597-618.

Limaiem, A. and ElMaraghy, H. E., 1997. A general method for analysing the accessibility of features using concentric spherical shells. *The International Journal of Advanced Manufacturing Technology*, 13(2), pp. 101-108.

Limaiem, A. and ElMaraghy, H. E., 1999. CATIP: A Computer-Aided Tactile Inspection Planning system. *International Journal of Production Research*, 37(2), pp. 447-465.

Lin, Y. J. and Mahabaleshwarkar, R., 1999. A generic algorithm for CAD-directed CMM dimensional inspection planning. *In Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, Detroit, USA, 2, pp. 1424-1429.

Lin, Y. J. and Murugappan, P., 1998. A new algorithm for CAD directed CMM dimensional inspection. *In Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, Leuven, Belgium, 1, pp. 893-898.

Liu, M., Liu, Y. and Ramani, K., 2009. Computing global visibility maps for regions on the boundaries of polyhedra using Minkowski sums. *Computer-Aided Design*, 41, pp. 668-680.

Liu, M. and Ramani, K., 2007. Computing an exact spherical visibility map for meshed polyhedra. *In: Proceedings of the 2007 ACM symposium on solid and physical modeling*(367-372).

Lovász, L., 1975. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13(4), pp. 383-390.

Lozano-Perez, T., 1983. Spatial Planning: A Configuration Space Approach. *IEEE Transactions on Computers*, C-32(2), pp. 108-120.

Lu, C. G., Morton, D., Wu, M. H. and Myler, P., 1999. Genetic algorithm modelling and solution of inspection path planning on a coordinate measuring machine (CMM). *International Journal of Advanced Manufacturing Technology*, 15, pp. 409-416.

Lu, E., Ni, J. and Wu, S. M., 1994. An algorithm for the generation of an optimum CMM inspection path. *Journal of Dynamic Systems Measurement and Control: Transactions of the ASME*, 116, pp. 396-404.

Lu, J. and Xie, M., 2010. Immune-genetic algorithm for traveling salesman problem. Traveling salesman problem, theory and applications. D. Davendra, InTech.

Lu, J. G., Ding, Y. L., Wu, B. and Xiao, S. D., 1996. An improved strategy for GAs in structural optimization. *Computers and Structures*, 61, pp. 1185-1191.

Maheshwari, A. and Wuhrer, S., 2009. Geodesic paths on 3D surfaces: survey and open problems. National Research Council Canada; NRC Institute for Information Technology.

Maisano, D. A., Jamshidi, J., Franchescini, F., Mastrogiacomo, L., Mileham, A. R., Owen, G. W. and Maropoulos, P. G., 2008. Indoor GPS: system functionality and initial performance evaluation. *International Journal of Engineering Manufacture*, 3(3), pp. 335-349.

Mankins, J. Z., 1995. Technology readiness levels: a white paper, National Aeronautics and Space Administration, USA.

Maropoulos, P. G., Guo, Y., Jamshidi, J. and Cai, B., 2008. Large volume metrology process models: a framework for integrating measurement with assembly planning. *Annals of the CIRP*, 57, pp. 477-480.

Martinez, D., Velho, L. and Carvalho, P. C., 2004. Geodesic paths on triangular meshes. *SIBGRAPI: Brazilian Symposium on Computer Graphics and Image Processing*, pp. 210-217.

Matai, R., Singh, S. P. and Mittal, M. L., 2010. Traveling salesman problem: an overview of applications, formulations, and solution approaches. Traveling salesman problem, theory and applications. D. Davendra, InTech.

MathWorks, Optimization Toolbox [online]. Available from: http://www.mathworks.co.uk/products/optimization/ [Accessed 05 January 2012]

Medland, A. J., Mullineux, G., Butler, C. and B. E. Jones, 1993. The integration of coordinate measuring machines within a design and manufacturing environment. *Proceedings of the Institution of Mechanical Engineers, Part B, Journal of Engineering Manufacture*, 207(B2), pp. 91-98.

Medland, A. J., Singh, R., Sittas, E. and Mullineux, G., 1990. Intelligent communication between CAD and manufacturing activities. *Proceedings of the 28th MATADOR Conference*, pp. 305-312.

Menon, M. M., Barry, R. E., Slotwinski, A., Kugel, H. W. and Skinner, C. H., 2001. Remote metrology, mapping, and motion sensing of plasma facing components using FM coherent laser radar. *Fusion Engineering and Design*, 58-59, pp. 495-498.

Merat, F. L., Radack, G. M., Roumina, K. and Ruegsegger, S., 1991. Automated inspection planning within the rapid design system. *Proceedings of IEEE International Conference on Systems Engineering*, pp. 42-48.

Mikhail, E. M., Bethel, J. S. and McGlone, J. C., 2001. Introduction to modern photogrammetry, John Wiley & Sons, New York.

Mishra, S., 1997. Visual Modeling & Unified Modeling Language (UML) : Introduction to UML, Rational Software Corporation [online]. Available from: http://www2.informatik.hu-berlin.de/~hs/Lehre/2004-WS_SWQS/20050107_Ex_UML.ppt [Accessed 25 February 2013]

Mitchell, J., Mount, D. and Papadimitriou, C., 1987. The discrete geodesic problem. *SIAM Journal on Computing*, 16, pp. 647-668.

MIL-STD 45662A, 1988. Military standard: calibration system requirements. .United States of America Department of Defence.

Mo, Y. B., 2010. The advantage of intelligent algorithms for TSP. Traveling salesman problem, theory and applications. D. Davendra, InTech.

Mohib, A., Azab, A. and ElMaraghy, H., 2009. Feature-based hybrid inspection planning: a mathmatical programming approach. *International Journal of Computer Integrated Manufacturing*, 22(2), pp. 13-29.

Muelaner, J. E., Cai, B. and Maropoulos, P. G., 2010. Large volume metrology instrument selection and measurability analysis. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 224(6), pp. 853-868.

Muelaner, J. E., Wang, Z., Jamshidi, J., Maropoulos, P. G., Mileham, A. R., Hughes, E. B. and Forbes, A. B., 2008. Uncertainty of angle measurement for a rotary-laser automatic theodolite (R-LAT). *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 223(B3), pp. 217-229.

Nemhauser, L. G. and Wolsey, L. A., 1988. Integer and combinatorial optimization. New York, John Wiley.

Nguyen, H. D., Yoshihara, I., Yamamori, K. and Yasunaga, M., 2007. Implementation of an effective hybrid GA for large-scale traveling salesman problems. *IEEE Transactions on Systems, Man and Cybernetics*, 37(1), pp. 92-99.

Nikon, 2011. iGPS [online]. Available from: http://www.nikonmetrology.com/en_eu/products/large-volume-applications/igps/igps [Accessed 17 September 2011]

Nikon, 2011. MV330/350 Laser Radar [online]. Available from: http://www.nikonmetrology.com/en_eu/products/large-volume-applications/laser-radar/mv330-mv350-laser-radar [Accessed 17 September 2011]

NRK, 2012. Spatial Analyzer [online]. Available from: http://www.kinematics.com/products/spatialanalyzer/overview.html [Accessed 15 March 2012]

O¨ncan, T., Altınel, I. K. and Laporte, G., 2009. A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers and Operations Research*, 36, pp. 637-654.

Obitko, M., 1998. Introduction to genetic algorithms [online]. Available from: http://www.obitko.com/tutorials/genetic-algorithms/index.php [Accessed 12 April 2011]

OMG, 2013, Unified Modelling Language [online]. Available from: http://www.uml.org/ [Accessed 25 February 2013]

Onoyama, T., Kubota, S., Oyanagi, K. and Tsuruta, S., 2000. A method for solving nested combinatorial optimization problems - a case of optimizing a large-scale distribution network. *Proceeding of IEEE SMC 2000*, Nashville, TN 1, pp. 340-345.

Onuut, S., Kara, S. S. and Isik, E., 2009. Long term supplier selection using a combined fuzzy mcdm approach: a case study for a telecommunication company. *Expert System Application*, 36(2), pp. 3887-3895.

Orman, A. J. and Williams, H. P., 2006. A survey of different integer programming formulations of the travelling salesman problem. *In: Kontoghiorghes E. & Gatu C. (eds). Optimisation, Econometric and Financial Analysis Advances in Computational Management Science*, Berlin, Heidelberg, 91-104.

Peggs, G. N., Maropoulos, P. G., Hughes, E. B., Forbes, A. B., Robson, S., Ziebart, M. and Muralikrishnan, B., 2009. Recent developments in large-scale dimensional metrology. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 223(6), pp. 571-595.

Potvin, J. V., 1996. Genetic algorithms for the traveling salesman problem. *Annals of Operations Research*, 63, pp. 339-370.

Puttock, M. J., 1978. Large-scale metrology. *Annals of the CIRP*, 27(1), pp. 351-356.

Qi, Y., Liu, F. and Jiao, L., 2008. Immune algorithm with selfadaptive reduction for large-scale TSP. *Journal of Software*, 19, pp. 1265-1273.

Roland, S., Ulrich, S. and Torsten, K., 1998. Integration of virtual reality based assembly simulation into CAD/CAM environments. *Proceedings of the 24th IEEE Industrial Electronics Conference*, Aachen, Germany, 4, pp. 2535-2537.

Sakurai, Y. and Tsuruta, S., 2010. A multi-world intelligent genetic algorithm to optimize delivery problem with interactive-time. Traveling salesman problem, theory and applications. D. Davendra, InTech.

Sen, S., 1993. Minimal cost set covering using probabilistic methods. *SAC '93 Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing: states of the art and practice*, pp. 157-164.

Sharir, M. and Schorr, A., 1986. On shortest paths in polyhedral spaces. *SIAM Journal on Computing*, 15, pp. 193-215.

Shorey, R., 2006. Mobile, wireless, and sensor networks: technology, applications, and future directions, John Wiley & Sons.

Spitz, S. N. and Requicha, A. A. G., 2000. Accessibility analysis using computer graphics hardware. *IEEE Transactions on Visualization and Computer Graphics*, 6(3), pp. 208-219.

Spitz, S. N. and Requicha, A. A. G., 2000. Multiple-goals path planning for coordinate measuring machines. *In Proceedings of ICRA '00 IEEE International Conference on Robotics and Automation*, San Francisco, CA , USA 3, pp. 2322-2327.

Spitz, S. N., Spyridi, A. J. and Requicha, A. A. G., 1999. Accessibility analysis for planning of dimensional inspection with co-ordinate measuring machines. *IEEE Transactions on Robotics and Automation*, 15(4), pp. 714-727.

Spyridi, A. J. and Requicha, A. A. G., 1990. Accessibility analysis for the automatic inspection of mechanical parts by coordinate measuring machines. *Proceedings of the IEEE International Conference on Robotics and Automation*, Cincinnati, OH, pp. 1284-1289.

Surazhsky, V., Surazhsky, T., Kirsanov, D., Gortler, S. J. and Hoppe, H., 2005. Fast exact and approximate geodesics on meshes. *ACM Transactions on Graphics*, 24(3), pp. 553-560.

Svestka, J. A. and Huckfeldt, V. E., 1973. Computational experience with an m-salesman traveling salesman algorithm. *Management Science*, 19(7), pp. 790-799.

Tannock, J. D. T., Cox, D. R., Lee, H. and SimsWilliams, J. H., 1993. Intelligent inspection planning and computer aided inspection. *Proceedings of the Institution of Mechanical Engineering, Part B: Journal of Engineering Manufacture*, 207, pp. 99-104.

Trafford, V., 2001. An unravelling of research methodology, Earlybrave Pulications Ltd.

Vafaeesefat, A. and ElMaraghy, H. A., 1998. Accessibility analysis in 5-axis machining of sculptured surfaces. *IEEE International Conference on Robotics and Automation*, Leuven , Belgium pp. 2464-2469.

Vafaeesefat, A. and ElMaraghy, H. A., 2000a. Automated accessibility analysis and measurement clustering for CMMs. *International Journal of Production Research*, 38(10), pp. 2215-2231.

Vafaeesefat, A. and ElMaraghy, H. A., 2000b. Optimal workpiece orientations for machining of sculptured surfaces. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 214(8), pp. 671-681.

Varadhan, G. and Manocha, D., 2006. Accurate Minkowski sum approximation of polyhedral models. *Graphical Models*, 68(4), pp. 343-355.

Walton, R. E., 1987. *Innovating to compete: lessons for diffusing and managing change in the workplace*, Jossey Bass Publishers.

Waurzyniak, P., 2003. Digital manufacturing taking hold. *Manufacturing Engineering Magazine*, 130(1), pp. 15-20.

Wikipedia, n. d.,Bounding volume. In: Wikipedia: the free encyclopaedia [online]. St Petersburg, Florida: Wikimedia Foundation. Available from: http://en.wikipedia.org/wiki/discrete_oriented_polytope [Accessed 20 November 2011]

Wikipedia, n. d.,Geometric dimensioning and tolerancing. In: Wikipedia: the free encyclopaedia [online]. St Petersburg, Florida: Wikimedia Foundation. Available from: http://en.wikipedia.org/wiki/greedy_algorithm [Accessed 18 November 2011]

Wikipedia, n. d.,Greedy algorithm. In: Wikipedia: the free encyclopaedia [online]. St Petersburg, Florida: Wikimedia Foundation. Available from: http://en.wikipedia.org/wiki/greedy_algorithm [Accessed 18 Decmber 2011]

Wikipedia, n. d.,STL (file format). In: Wikipedia: the free encyclopaedia [online]. St Petersburg, Florida: Wikimedia Foundation. Available from: http://en.wikipedia.org/wiki/stl_(file_format) [Accessed 20 November 2011]

Williamson, D. P., 1998. Lecture Notes on Approximation Algorithms. IBM Research Report. IBM Research Division TJ Watson Research Center, New York.

Wong, F. S. Y., Chuah, K. B. and Venuvinod, P. K., 2006. Automated inspection process planning: algorithmic inspection feature recognition and case representation for CBR. *Robotics and Computer-Integrated Manufacturing*, 2, pp. 56-68.

Woo, T. C., 1994. Visibility maps and spherical algorithms. *Computer-Aided Design*, 26(1), pp. 6-16.

Wu, Y., Liu, S. and Zhang, G., 2004. Improvement of co-ordinate measuring machine probing accessibility. *Precision Engineering*, 28(1), pp. 89-94.

Xin, S. Q. and Wang, G. J., 2007. Efficiently determining a locally exact shortest path on polyhedral surfaces. *Computer-Aided Design*, 39(12), pp. 1081-1090.

Xu, Z. S., 2007. Intuitionistic fuzzy aggregation operator. *IEE Transaction of Fuzzy Systems*, 15(6), pp. 1179-1187.

Xu, Z. S., 2007. Intuitionistic preference relations and their application in group decision making. *Information Sciences*, 177, pp. 2363-2379.

Yang, C. C., Marefat, M. M. and Kashyap, R. L., 1994. Active visual inspection based on cad models. *Proceedings of IEEE: Robotics and Automation*, 2, pp. 1120-1125.

Yang, W., Ding, H. and Xiong, Y., 1999. Manufacturability Analysis for a Sculptured Surface Using Visibility Cone Computation. *The International Journal of Advanced Manufacturing Technology*, 15(5), pp. 317-321.

Yau, H. T. and Menq, C. H., 1991. Path planning for automated dimensional inspection using coordinate measuring machines. *Proceedings of the 1991 IEEE International Conference on Robotics and Automation,*, pp. 1934-1939.

Yau, H. T. and Menq, C. H., 1991. Path planning for automated dimentional inspection using coordinate measuring machines. *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pp. 1934-1939.

Yau, H. T. and Menq, C. H., 1992. An automated dimensional inspection environment for manufactured parts using coordinate measuring machines. *International Journal of Production Research*, 30(7), pp. 1517-1536.

Yau, H. T. and Menq, C. H., 1992. An intelligent planning environment for automated dimensional inspection using coordinate measuring machines. *ASME Journal of Engineering for Industry*, 114, pp. 222-230.

Yin, R., 1993. Applications of case study resarch, Sage.

Yin, Z.-P., Ding, H. and Xiong, Y.-L., 2000. Visibility theory and algorithms with application to manufacturing processes. *International Journal of Production Research*, 38(13), pp. 2891-2909.

Yin, Z.-P., Ding, H. and Xiong, Y.-L., 2002. Accessibility analysis in manufacturing processes using visibility cones. *Science in China*, 45(1), pp. 47-57.

Zadeh, L. A., 1965. Fuzzy sets. *Infomation and Control*, 8, pp. 338-353.

Zeng, C. and Gu, T., 2007. A novel immunity-growth genetic algorithm for traveling salesman problem. *Proceedings of Third International Conference on Natural Computation, ICNC 2007*, Haikou, Hainan, China, pp. 394-398.

Zhang, S. G., Ajmal, A., Wootton, J. and Chisholm, A., 2000. A feature-based inspection process planning system for co-ordinate measuring machine (CMM). *Journal of Materials Processing Technology*, 107(1), pp. 111-118.

Zhao, F., Xu, X. and Xie, S. Q., 2009. Computer-aided inspection planning- the state of the art. *Computers in Industry*, 60, pp. 453-466.

Ziemian, C. W. and Medeiros, D. J., 1997. Automated feature accessibility algorithm for inspection on a coordinate measuring machine. *International Journal of Production Research*, 35(10), pp. 2839-2856.

Ziemian, C. W. and Medeiros, D. J., 1998. Automating probe selection planning for inspection on measuring machine and part setup a coordinate. *International Journal of Computer Integrated Manufacturing*, 11(5), pp. 448-460.